Discontinuous Galerkin Level Set Method for Interface Capturing

Stanley Osher*and Jue Yan†

November, 2005.

Abstract

In this paper, we combine a high-order Discontinous Galerkin (DG) method and level set method solving the interface problem in a complex incompressible flow. The scheme is L^2 stable and conservative. It improves the mass conservative property of the level set method. Numerical examples demonstrate the high order accuracy of the method and the high resolution especially when the interface undergoes large topological changes. Local level set technique is applied to improve the efficiency of the method.

Key Words: Discontinuous Galerkin method, level set method.

*Department of Mathematics, UCLA, Los Angeles, CA 90095; sjo@math.ucla.edu
†Department of Mathematics, UCLA, Los Angeles, CA 90095; yan@math.ucla.edu

1 Introduction

A large class of fluid problems involve moving interfaces. Applications include multi-phase flow, multi-medium flow, wave and bubble dynamics, biological flow coupled to deformable tissue/elastic structure and detonation problems. Most interface modeling can be classified into interface tracking or interface capturing methods. For interface tracking method, the interface is explicitly tracked through the trajectories of fluid particles in a Lagrangian manner, see [6] by Glimm and [19] by Tryggvason. For interface capturing method the interface is represented as either a discontinuous characteristic function(VOF method), or zero level set of an implicit continuous function (level set method).

Level set method for capturing moving fronts was first introduced by Osher and Sethian [9] in 1987. Over the years, the method has proven to be phenomenally successful in a wide variety of applications, including fluid dynamics, material sciences, computer vision and image processing. We refer to the book [8] for an extensive exposition of the level set method.

The idea of level set method is to use the zero level set of a smooth function $\phi(x,t)$ (at least Lipschitz continuous) to implicitly represent the moving interface, as $\Gamma(t) = \{(x,y) \in R^2 \,|\, \phi(x,y,t) = 0\}$. This implicit Eulerian type of representation gives level set method many attractive advantages: 1) it's easy to extend from 2D to 3D; (not the case for front tracking) 2) it can automatically handle topological changes such as merging and breaking; 3) it's easy to compute geometric quantities such as normals and curvatures. Almost from their inception, level set methods have been used to model multi-phase immiscible incompressible flows [17]. Unfortunately, the discretization of the level set equation can lead to significant numerical dissipation that usually manifests itself as a loss of mass(or volume). In fact, this has been the major criticism levied against level set methods for incompressible flow. Currently, there are still a large portion of level set research work be dedicated to improving the mass conservations, e.g. [4].

Level set equation essentially belongs to the class of Hamilton-Jacobi equations. Solutions to Hamilton-Jacobi equations are typically continuous but may develop singularities in the derivatives even if the initial condition is smooth. There have been intensive studies with finite difference methods for solving Hamilton-Jacobi equations, from the first order monotone scheme [3] in 1984 by Crandall and Lions, to the second order finite difference ENO scheme by Osher and Sethian in [9]. A general framework for higher-order ENO scheme was given by Osher and Shu in [14], and extension to a higher-order WENO scheme was proposed by Jiang and Peng in [7]. Compared to finite difference ENO/WENO schemes [7] and [18], the discontinuous Galerkin method can better resolve the kink corners with discontinuous derivatives even after a long term run. To resolve the "mass loss" issue for the level set method, we will solve the level set equation with discontinuous Galerkin method.

The discontinuous Galerkin method is a class of finite element methods using completely discontinuous piecewise polynomial approximations for space discretizations, then coupling with nonlinear stable high-order Runge-Kutta methods for time discretization. Recently discontinuous Galerkin methods become highly attractive and popular, mainly because these methods are high-order accurate, nonlinear stable, highly parallelizable, easy to handle complicated geometries and boundary conditions, and capable to capture discontinuities without spurious oscillations.

A major development of the discontinuous Galerkin method was carried out by Cockburn

et al. for hyperbolic conservation laws in the past decade (refer to the review paper[2]). The use of completely discontinuous approximations gives DG methods several advantages: 1) it can be easily designed for any order of accuracy, thus allowing for efficient p-adaptivity; 2) it can be used on arbitrary triangulations, thus allowing for efficient h-adaptivity; 3) it is highly parallelizable; (The mass matrix is block diagonal and easily invertible) 4) it has excellent provable nonlinear stability.

For discontinuous Galerkin methods, one nice property among many others is its small numerical dissipation when applying with high order polynomial approximations. It is numerically observed by Hu and Atkins in [5], and later theoretically proved by Ainswroth in [1] that the dissipation errors decay at the order of $(2k + 2)$ when P^k polynomials are used in spacial approximations.

In this paper, we apply high-order discontinuous Galerkin methods (P^k polynomials with $k \geq 2$) to solve level set equations for a set of test problems which approximate flows with large vortical components. Numerical examples demonstrate that discontinuous Galerkin method coupled with level set methods conserves the mass very well, even after long term run and when the interface undergoes large topological changes.

Discontinuous Galerkin method has the flexibility to apply different base functions on different computational cells, for example the so-called $p-$ adaptivity. A local level set technique or $p-$adaptivity is applied to improve the efficiency of the method. We dynamically apply high order polynomial approximations around the interface (zero level set), and apply low order polynomial approximations away from the interface.

The organization of the paper is as follows. In Section 2, we review the level set method. In Section 3, we describe the formulation of the discontinuous Galerkin method and the $p-$adaptive local level set technique. L^2 stability and the corresponding L^2 error estimate is discussed. Numerical examples are presented in Section 4.

2 Level set method

Suppose we will describe a moving curve $\Gamma(t)$ in 2D (or a surface in 3D) that bounds a region $\Omega \in R^2$ (not necessary to be closed) as in figure 1. The motion of the curve (interface) is determined by a velocity field $\mathbf{u} = (u_1, u_2)$, which may depend on the position, time, the geometry of the interface like curvature or be given externally, e.g. the material velocity of the fluid flow.

The underlying idea behind level set methods is to introduce a level set function $\phi(x, y, t)$ (at least Lipchitzs continuous) with one dimensional higher, which has the property that is positive in one region and negative in the other, with the zero level contour of $\phi(x, y, t)$ always representing the current position of the moving interface $\Gamma(t) = \{(x, y)|\phi(x, y, t) = 0\}$.

The evolution equation for the level set function is given by

$$\phi_t + \mathbf{u} \cdot \bigtriangledown \phi = 0 \tag{1}$$

It is convenient to make ϕ as a signed distance function to the interface so that $|\bigtriangledown \phi| = 1$. This ensures that the level set is a smoothly varying function well suited for high order accurate numerical methods. We use the following steady state solution to obtain the initial condition

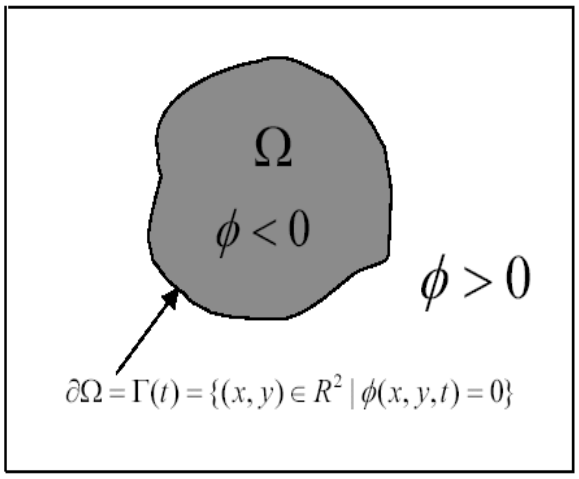

Figure 1: Explanation of Level Set Method

of the level set distance function,

$$\phi_t + S(\phi_0)(|\bigtriangledown \phi| - 1) = 0, \tag{2}$$

Where $S(\phi) = \frac{\phi}{\sqrt{(\phi^2 + |\bigtriangledown\phi|^2(\Delta x)^2)}}$

This Hamilton-Jacobi equation is mainly used as re-initialization in most level set related applications when the level set function cease to be a signed distance function. we should mention we do not apply this re-initialization step in our numerical simulations. We use Godnov's scheme to discretize the $S(\phi_0)|\bigtriangledown \phi|$ hyperbolic term. After finding a numerical approximation to this term, we could combine it with Runge-Kutta method for time discretization.

One major advantage of level set method is the geometric quantities can be easily calculated with the level set function,e.g. the unit normal,

$$\vec{N} = \frac{\bigtriangledown\phi}{|\bigtriangledown \phi|}$$

and the curvature,

$$\kappa = \bigtriangledown \cdot (\frac{\bigtriangledown\phi}{|\bigtriangledown \phi|}).$$

In this paper, the velocity field $\mathbf{u} = (u_1(x, y), u_2(x, y))$ is given externally, thus we directly discretize (1) with discontinuous Galerkin method and capture the movement of the interface with the zero level set.

3 Discontinuous Galerkin Method for level set equation

3.1 Discontinuous Galerkin method

We are given a 2-D incompressible flow with velocity field $\mathbf{u} = (u_1(x,y), u_2(x,y))$. With the incompressible property,

$$\nabla \cdot \mathbf{u} = 0,$$

we rewrite the level set equation (1) as a variable coefficient conservation law as,

$$\phi_t + \nabla \cdot (\mathbf{u}\phi) = 0, \quad (x,y) \in \Omega. \tag{3}$$

Now we formulate the discontinuous Galerkin finite element method solving (3). Initial data is given as $\phi(x,y,0) = \phi_0(x,y)$ and we have Dirichlet type or periodic boundary condition.

Let's have a regular partition over the computational domain Ω, $\mathcal{T}_{\Delta x} = \{K\}$. The piecewise polynomial space is defined as:

$$\mathcal{V}_{\Delta x} == \left\{ v \in L^2(\Omega) : \ v|_K \in P^k(K) \quad \text{for } K \in \mathcal{T}_{\Delta x} \right\},$$

where $P^k(K)$ is the space of polynomial with degree k on element K.

We seek a discontinuous piecewise polynomial approximation ϕ, such that at each time t, $\phi(x,y,t) \in \mathcal{V}_{\Delta x}$. We mention that we abuse the notation here with ϕ representing both the exact solution and the numerical solution. In a word, restricted on each element K of the partition $\mathcal{V}_{\Delta x}$, the numerical solution ϕ is a piecewise polynomial with degree at most k. Now we multiply the equation (3) by test function $\nu \in \mathcal{V}_{\Delta x}$, integrate over the element K, have the integration by parts and we obtain,

$$\int_K \phi_t \nu dxdy - \int_K (\mathbf{u}\phi) \cdot \nabla \nu dxdy + \int_{\partial K} \widehat{\mathbf{u}\phi}_{n_K} \nu ds \ = 0, \tag{4}$$

with $\widehat{\mathbf{u}\phi}_{n_K}$ as the numerical flux.

To complete the definition of the discontinuous Galerkin method, it only remains to define the numerical flux term $\widehat{\mathbf{u}\phi}_{n_K}$. Here n_K denotes the outward unit normal for element K along the element boundary ∂K. We take the numerical flux $\widehat{\mathbf{u}\phi}_{n_K}$ to be solely a function of the traces ϕ^{int_K} and ϕ^{ext_K}. Here ϕ^{int_K} denotes the value of ϕ evaluated from the inside of element K, and ϕ^{ext_K} denotes the value of ϕ evaluated from the outside of element K (inside of its neighboring element K').

We take $\widehat{\mathbf{u}\phi}_{n_K}$ as a monotone numerical flux to approximate $\mathbf{u}\phi_{n_K} = \mathbf{u}\phi \cdot n_K$ at the discontinuous element boundary. Namely $\widehat{\mathbf{u}\phi}_{n_K}$ is a Lipschitz continuous function in both arguments ϕ^{int_K} and ϕ^{ext_K}, is consistent with $\mathbf{u}\phi_{n_K}$ and is non-decreasing for argument ϕ^{int_K} and non-increasing for argument ϕ^{ext_K}. Moreover, it is a uniquely defined quantity on the element boundary ∂K (that is, there is only one flux defined at each edge shared by two elements K and its neighbor K'). Thus it is conservative. For example, we have the following upwinding/Lax-Friedrichs numerical flux used in our simulations,

$$\widehat{\mathbf{u}\phi}_{n_K} = \frac{1}{2}\Big(\sum_{i=1}^{2} u_i(\phi^{int_K} + \phi^{ext_K}) n_{i,K} - \alpha(\phi^{ext_K} - \phi^{int_K}) \Big) \tag{5}$$

where

$$\alpha = \max_{(x,y)\in S}|u_1 n_{1,K} + u_2 n_{2,K}|.$$

With the local or global domain S to evaluate the maximum, correspondingly we have the local or global Lax Friedrichs coefficient α. Now let's consider the theoretical properties of the above discontinuous Galerkin method (4)-(5).

Corollary 3.1 (Locally Conservative) *The scheme (4)-(5) is locally conservative, that is,*

$$\int_K \phi_t dxdy + \int_{\partial K} \widehat{\mathbf{u}\phi}_{n_K} d\Gamma = 0.$$

Proof. This property is easily obtained from equation (4) by taking the test function $\nu = 1$ as a constant.

Corollary 3.2 (L^2 stability) *The discontinuous Galerkin method (4)-(5) is L^2 stable,*

$$\frac{1}{2}\frac{d}{dt}\int_\Omega \phi^2(x,y,t)dxdy \le 0.$$

Proof.
Since (4) holds for any test function in the solution space $\mathcal{V}_{\Delta x}$, let's particularly take $\nu = \phi$. With $\nu = \phi$ in (4) we have,

$$\int_K \phi_t \phi \, dxdy - \int_K (\mathbf{u}\phi)\cdot\nabla\phi \, dxdy + \int_{\partial K} \widehat{\mathbf{u}\phi}_{n_K} \phi^{int_K} \, ds = 0. \tag{6}$$

Using the incompressible property $\nabla \cdot \mathbf{u} = 0$, we have

$$\begin{aligned} \textstyle\int_K (\mathbf{u}\phi)\cdot\nabla\phi dxdy &= \textstyle\int_K \nabla\cdot(\frac{1}{2}\mathbf{u}\phi^2)dxdy \\ &= \textstyle\int_{\partial K}(\frac{1}{2}\mathbf{u}\phi^2)\cdot n_K ds. \end{aligned}$$

Now we have (6) as,

$$\int_K \phi_t\phi \, dxdy - \int_{\partial K}(\frac{1}{2}\mathbf{u}\phi^2)\cdot n_K \, d\Gamma + \int_{\partial K} \widehat{\mathbf{u}\phi}_{n_K}\phi^{int_K} \, ds = 0. \tag{7}$$

Sum (7) over all elements $K \in \mathcal{T}_{\Delta x}$, we have

$$\frac{1}{2}\frac{d}{dt}\int_\Omega \phi^2 \, dxdy + I(\phi) = 0, \tag{8}$$

with

$$I(\phi) \;= \textstyle\sum_e \int_e \widehat{\mathbf{u}\phi}_{n_K}(\phi^{int_K} - \phi^{ext_K}) \, ds - \sum_e \int_e (\frac{1}{2}\mathbf{u}(\phi^{int_K})^2 - \frac{1}{2}\mathbf{u}(\phi^{ext_K})^2)\cdot n_K \, ds.$$

Here the edge $e = K \cap K'$ is shared by element K and K', representing the collection of all element boundaries. We should mention we consider the periodic boundary condition case. We know $n_{K'} = -n_K$. With the consistency and the monotonicity of the numerical flux we have,

$$I(\phi) \ge 0.$$

From (8), we easily obtain that the L^2 energy is non-increasing with time evolution.

Proposition 3.1 (error estimate) *Let's use e_ϕ denote the error between the exact solution of (1) and the DG solution of (4) - (5). We have*

$$\|e_\phi\|_{L^2} \leq C\Delta x^{k+1/2}, \tag{9}$$

where the constant C depends on the regularity of the exact solution ϕ.

We refer to [2] for the reference of the proof of the above error estimate.

Up to now, we have taken the method of lines approach and have left time variable t continuous. For time discretization we use the strong-stability preserving (SSP) explicit high order TVD Runge-Kutta method [13, 12] to match the accuracy in space. For example, the third order SSP Runge-Kutta method is

$$\begin{cases} \phi^{(1)} = \phi^n + \Delta t H(\phi^n) \\ \phi^{(2)} = \frac{3}{4}\phi^n + \frac{1}{4}(\phi^{(1)} + \Delta t H(\phi^{(1)})) \\ \phi^{n+1} = \frac{1}{3}\phi^n + \frac{2}{3}(\phi^{(2)} + \Delta t H(\phi^{(2)})) \end{cases}$$

where $H(\phi)$ is the spatial operator with the discontinuous Galerkin discretization in space.

3.2 Local Level Set Technique

Now we consider $p-$adaptive local let set technique to obtain a more efficient discontinuous Galerkin method (4).

The discontinuous Galerkin solution space $\mathcal{V}_{\Delta x}$ consists of discontinuous piecewise polynomials, thus the DG method has the nice adaptive property that we can apply different degrees of polynomials on different elements. For interface capturing problem, we are mainly interested in the position of the zero level set. We can use higher order polynomial approximations around the zero level set to obtain the high order accuracy and the high resolution, and use lower order polynomials away from the zero level set. As displayed in Figure 2, we apply P^1 linear polynomial approximations two or three cells away from the zero level set. In the implementation of the local level set technique, we have a flag to mark the location of the zero level set. Near the zero level set (dark cells in Figure 2) we apply P^2 or P^3 polynomials approximations and away from the zero level set we use P^1 polynomials. We dynamically update the flag set to mark the zero level set to implement the $p-$adaptivity of the scheme. This technique dramatically increases the efficiency of the DG scheme.

4 Numerical experiments

In this section, we implement the discontinuous Galerkin level set methods to a couple of test problems, from rigid body rotation in an incompressible flow to complex vortex flows.

Example I: A circular disk rigid body rotation in a constant vorticity velocity field

A circular disk is put inside a box with a constant vorticity velocity field. The circular disk simply rotates around the box center. After one period the disk should goes back to its original position. For the circle shape interface, we can write down the exact solution of the level set

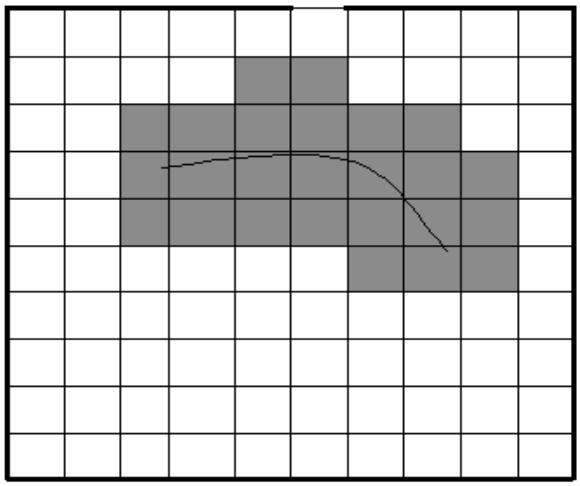

Figure 2: Explanation of Local Level Set Technique

k		**N=40**	**N=80**		**N=120**		**N=160**	
		error	error	order	error	order	error	order
1	L^1	0.13214e-02	0.12993e-03	3.34	0.33869e-04	3.31	0.16036e-04	2.59
	L^∞	0.52365e-02	0.19236e-02	1.44	0.54360e-03	3.11	0.16685e-03	4.10
2	L^1	0.12348e-03	0.23593e-02	3.70	0.16718e-05	4.27	0.51639e-06	4.08
	L^∞	0.23593e-02	0.23593e-02	1.87	0.11577e-03	4.23	0.37316e-04	3.93
3	L^1	0.42471e-04	0.27473e-05	3.95	0.42860e-06	4.58	0.94291e-07	5.26
	L^∞	0.16750e-02	0.25007e-03	2.74	0.58097e-04	3.59	0.12164e-04	5.43

Table 1: Accuracy test for a circle rigid body rotation in a constant vorticity velocity field. L^1 and L^∞ errors are computed after one period rotation. Global Lax-Friedrich numerical flux is used. The accuracy is computed in the smooth region $\|\phi\| \leq 0.05$.

function (the distance to the circle boundary). This example is used to test the accuracy of the discontinuous Galerkin method.

Computational domain is set on $[0,1]$x$[0,1]$. Initial data is set as the distance function to the circle centered at (0.5,0.75) with a radius of 0.15. The velocity field is given as,

$$\begin{cases} u_1(x,y) = \frac{\pi}{3.14}(0.5-y) \\ u_2(x,y) = \frac{\pi}{3.14}(x-0.5). \end{cases} \tag{10}$$

In Table 1, we check the L^1 and L^∞ errors after one period of rotation in domain $|\phi| \leq 0.05$, in which the solution is smooth. Here we use global Lax-Friedrich flux, and $(k+1)$th order of accuracy is obtained with P^k polynomials aprroximations. We also test it with upwind numerical flux, and obtain similar accuracy results which is listed in Table 2. Notice the errors are computed in the smooth region $[0.3, 0.7]$X$[0.5, 0.7]$.

Example II: Zalesak's Disk in a constant vorticity velocity field

An acceptable interface capturing method must translate and rotate fluid body without significant distortion or degradation of fluid interfaces. Mass (volume of the rigid body) should

k		N=40	N=80		N=120		N=160	
		error	error	order	error	order	error	order
1	L^1	0.259e-03	0.488e-04	2.40	0.199e-04	2.20	0.112e-04	2.00
	L^∞	0.344e-02	0.102e-02	1.76	0.337e-03	2.72	0.176e-03	2.26
2	L^1	0.106e-04	0.806e-06	3.72	0.212e-06	3.29	0.882e-07	3.04
	L^∞	0.449e-03	0.594e-04	2.91	0.192e-04	2.78	0.912e-05	2.60
3	L^1	0.206e-05	0.600e-07	5.09	0.620E-08	5.59	0.155e-08	4.82
	L^∞	0.149e-03	0.762e-05	4.28	0.981E-06	5.05	0.289E-06	4.25

Table 2: Same test problem as the one in Table 1. Upwind numerical flux is used and the accuracy is obtained in the smooth region $[0.3, 0.7]X[0.5, 0.7]$.

k		N=40	N=80		N=120		N=160	
		error	error	order	error	order	error	order
2	L^1	0.525e-03	0.529e-04	3.31	0.125e-04	3.56	0.535e-05	2.94
	L^∞	0.274e-02	0.827e-03	1.72	0.792e-03	0.10	0.604e-03	0.94
3	L^1	0.265e-03	0.251e-04	3.40	0.641e-05	3.36	0.310e-05	2.52
	L^∞	0.181e-02	0.643e-03	1.50	0.399e-03	1.17	0.360E-03	0.38

Table 3: An accuracy test for the zalesak disk in the constant vorticity velocity field. Upwind flux is used and accuracy check is obtained in smooth region $[0.3, 0.4]X[0.6, 0.9]$.

also be conserved rigorously. In this example, we use the same velocity field as (10) in Example I. The rigid body is a circle with a rectangular notch, see Figure 3. After one period rotation, the notched circle should go back to its original position. With a mesh setup as $N \times N = 100 \times 100$, there are only 5 cells in each direction inside the rectangular notch. This is a nontrivial test. It's hard to keep the sharp corners and preserve the total mass (volume). In paper [4], Enright et al. tested this case with Hamilton-Jacobi high order WENO5 finite difference scheme, and they show in Figure 15 of [4] that the numerical solution easily smeared out the notch shape after one rotation and it gets worse after 2 rotations.

We first test the L^1 and L^∞ errors (after one period rotation) in the region where the exact solution is smooth and can be obtained. The results are listed in Table 3. The initial zelasak disk is shown in Figure 3. Figure 4 shows the DG numerical approximation with P^2 polynomials and p^3 polynomials approximation is shown in Figure 5. Both are tested with mesh size 100×100. A zoomed-in figure is shown in Figure 6. To check the robustness of the DG scheme, we run the code after ten periods of rotations, and the result is shown in figure 7. We see the discontinuous Galerkin method can resolve the corners sharply, even after a long term run.

Example III: a circular fluid body in a non-constant vortex velocity field

The computational domain is set as $[0, 1] \times [0, 1]$, initial data is a circle centered at (0.5,0.75) with a radius of 0.15. The velocity field is the given as below,

$$\begin{cases} u_1(x, y) = -\sin(2\pi y)\sin^2(\pi x) \\ u_2(x, y) = \sin(2\pi x)\sin^2(\pi y). \end{cases} \tag{11}$$

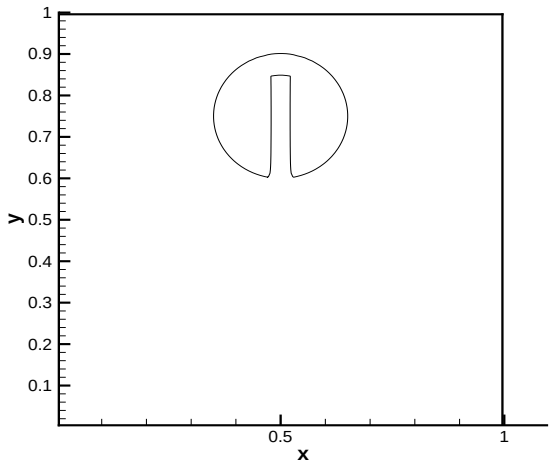

Figure 3: Initial zalesak disk in a box

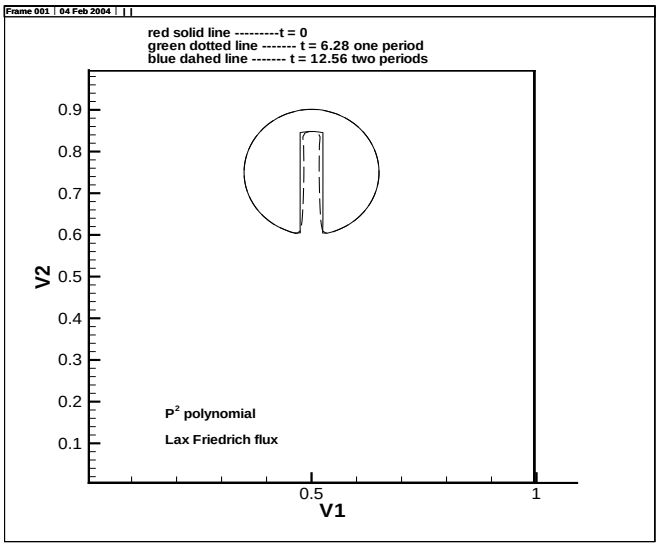

Figure 4: Zalesak Disk: Lax Friedrich flux with P^2 polynomial and mesh size 100x100.

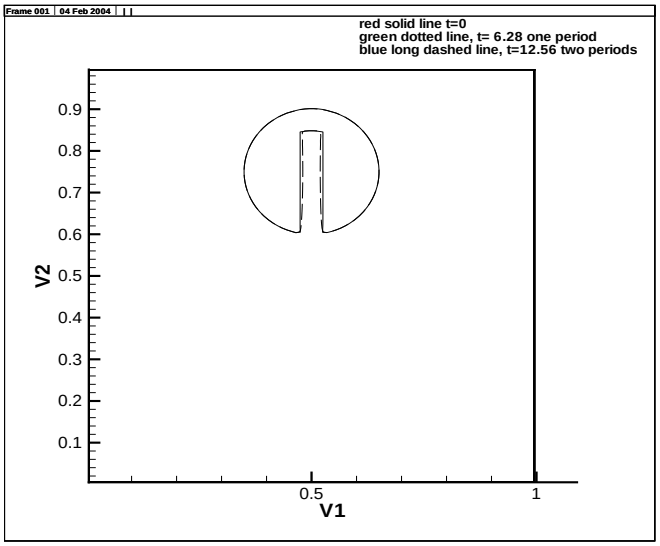

Figure 5: Zalesak Disk: upwinding flux with P^3 polynomial and mesh size 100x100.

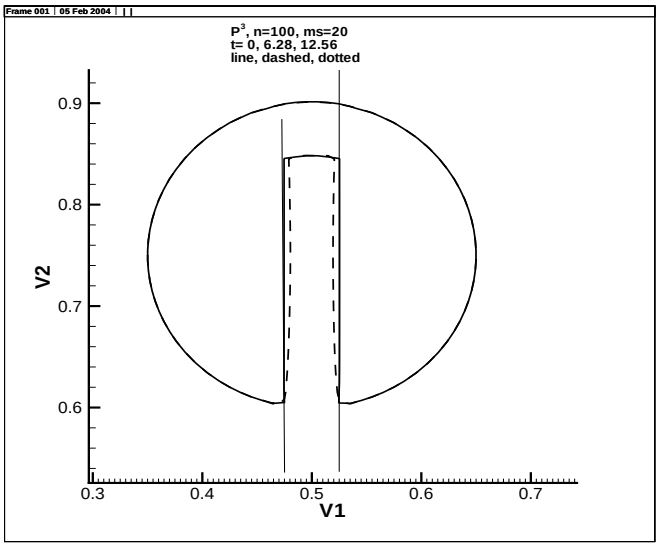

Figure 6: Zalesak Disk: upwind flux with P^3 polynomial and mesh size 100x100. Zoom in

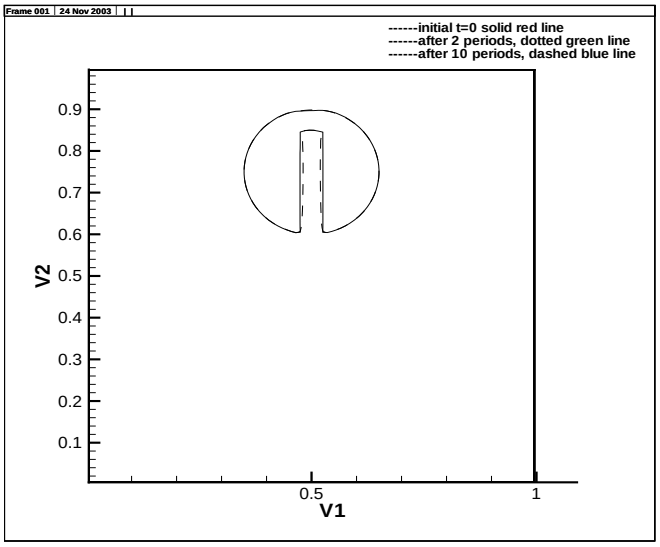

Figure 7: Zalesak Disk: upwind flux with P^3 polynomial and mesh size 100x100. after ten periods rotations.

k		**N=50X50**	**N=80X80**		**N=100X100**		**N=120X120**	
		error	error	order	error	order	error	order
2	L^∞	2.3554e-03	7.8355e-04	2.34	5.5529e-04	3.56	3.2123e-04	2.87

Table 4: Single vortex: the area loss after the time reversal(at $T = 6$) with P^2 polynomial approximations.

Under this non-constant vorticity velocity field, the fluid body (circle) will be spun toward the vortex center, and be stretched and torn into very thin fluid filament. This example tests whether or not the numerical method is robust when the interface undergoes gross topology change. In figure 8, we compute the DG approximations up to final time $t = 5$. We observe that the DG smethod can capture very thin fluid interface almost with mesh size large. To further demonstrate the mass conservation property of the DG method, we apply a time-reversal function to the velocity field, such that after one period $T = 6$, the fluid body will go back to its original circular shape, as shown in figure 9. We compute the area loss (L^∞ error) with refined mesh, and we obtain a nearly third order of accuracy with p^2 polynomial approximations. Results are listed in table 4.

Example IV: Multiple Vortex

The computational domain is set as $[0, 1] \times [0, 1]$, initial data is a circled body centered at (0.5,0.75) with a radius of 0.15. The velocity field is given as ,

$$\begin{cases} u_1(x, y) = \sin(4\pi(x + 0.5))\sin(4\pi(y + 0.5)) \\ u_2(x, y) = \cos(4\pi(x + 0.5))\cos(4\pi(y + 0.5)). \end{cases} \tag{12}$$

We see we have 16 non-constant vortexes in this velocity field.

We compute this example with P^2 polynomials and show the result at $T = 1$ in Figure 10.

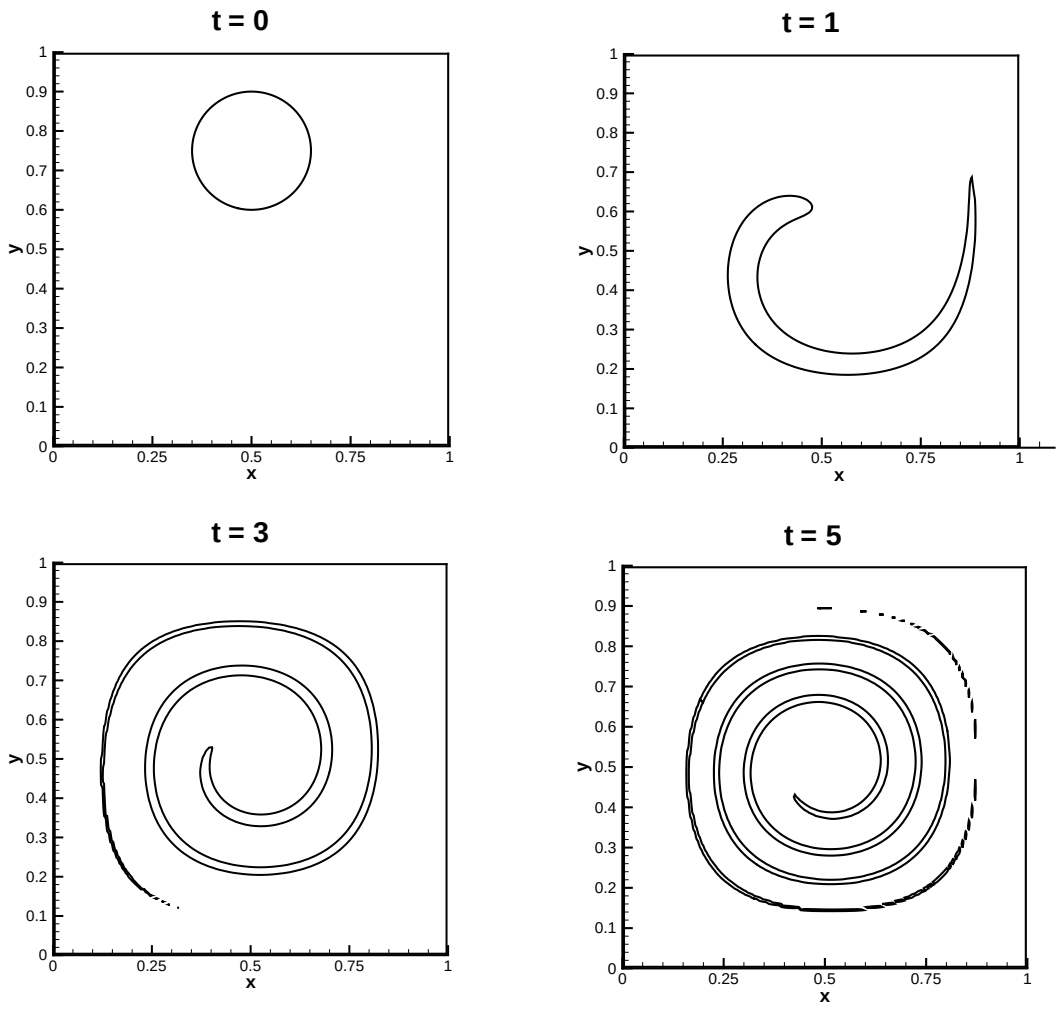

Figure 8: Single vortex: Lax Friedrich flux. P^3 polynomial. Mesh: 128x128

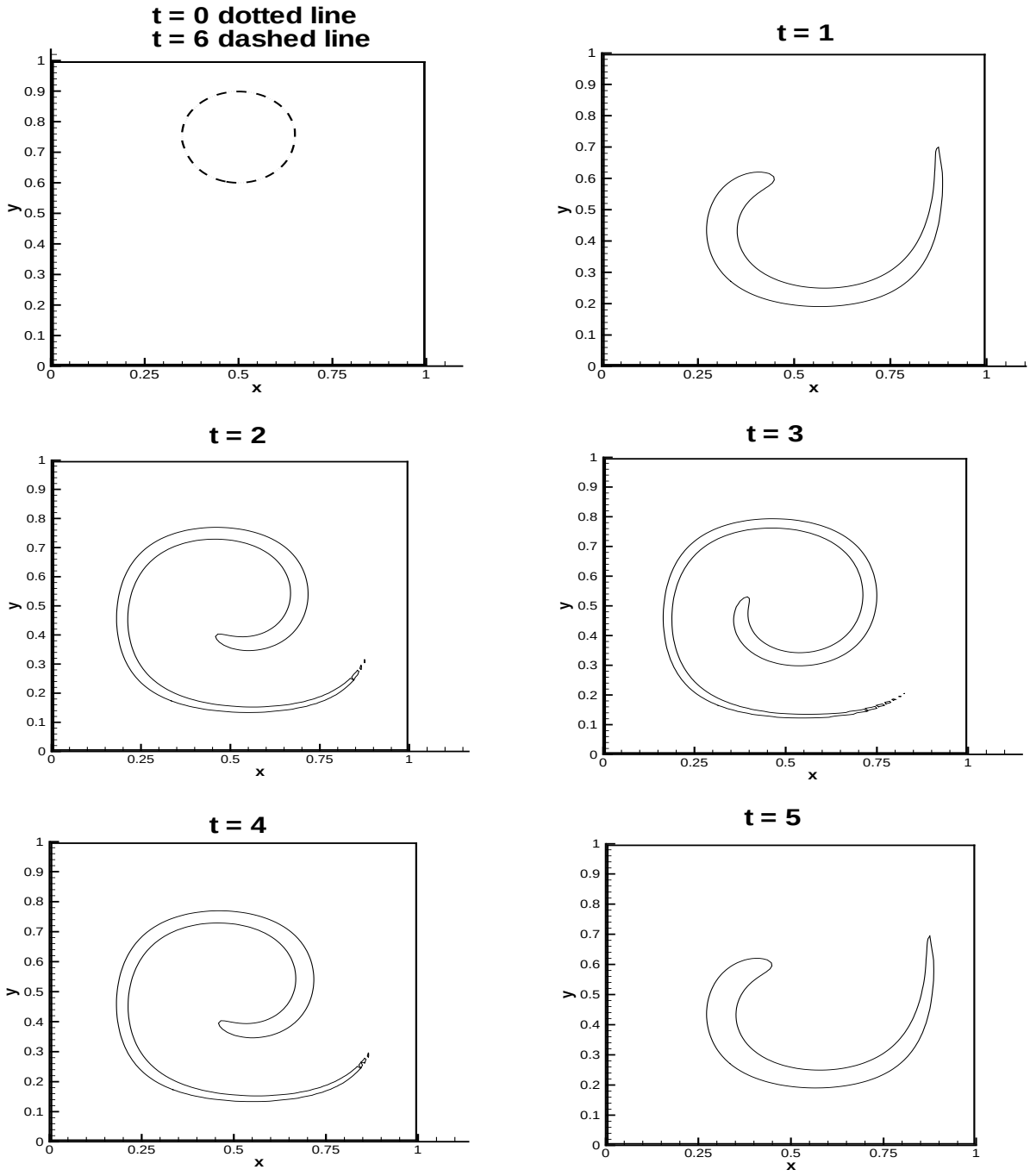

Figure 9: Single vortex: time reversal at $T = 6$. P^2 polynomial. Mesh: 100x100

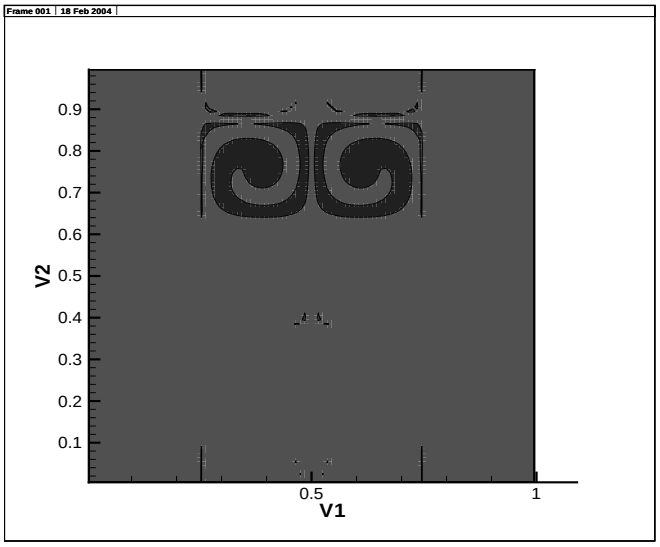

Figure 10: 16 vortexes: Lax Friedrich flux with P^2 polynomial with mesh size 128x128.

Again the DG solution can resolve the very thin fluid filament. Our result is similar to those in literature, see [4].

References

[1] Mark Ainsworth, *Dispersive and dissipative behavior of high order discontinuous Galerkin finite element methods*, J. Comput. Phys., **198**,(2004), pp.106–130.

[2] B. Cockburn and C.-W. Shu, *Runge-Kutta discontinuous Galerkin methods for convection-dominated problems*, J. Sci. Comput., 16(3):173–261, 2001.

[3] M. G. Crandall and P. L. Lions, *Two approximations of solutions of Hamilton-jacobi equations*, math. Comp., **43** (1984), pp. 1-19.

[4] D. Enright, R. Fedkiw, J. Ferziger, and I. Mitchell, *A hybrid particle level set method for improved interface capturing*, J. Comput. Phys., 183 (2002), no. 1, 83–116.

[5] F. Q. Hu and H. L. Atkins, *Eigensolution analysis of the discontinuous Galerkin method with n onuniform grids: I. one space dimension*, J. Comput. Phys., **198**,(2004), Pages 106-130.

[6] J. Glimm, J. W. Grove, X. L. Li, and N. Zhao, *Simple front tracking*, Contemp. Math., 238:133–149, 1999.

[7] G.-S. Jiang and D. Peng, *Weighted ENO schemes for Hamilton-Jacobi equations*, SIAM J. Sci. Comput., 21 (2000), no. 6, 2126–2143.

[8] S. Osher and R. Fedkiw, *Level set methods and dynamic implicit surfaces*, Applied Mathematical Sciences, 153. Springer-Verlag, New York, 2003.

[9] S. Osher and J. Sethian, *Fronts propagating with curvature-dependent speed: algorithms based on Hamilton-Jacobi formulations*, J. Comput. Phys., 79 (1988), no. 1, 12–49.

[10] J. Qiu, B.C. Khoo and C.-W. Shu, *A numerical study for the performance of the Runge-Kutta discontinuous Galerkin method based on different numerical fluxes*, Journal of Computational Physics, to appear.

[11] J. Sethian and P. Smereka, *Level set methods for fluid interfaces*, Annual review of fluid mechanics, Vol. 35(2003), 341–372.

[12] C.-W. Shu and S. Osher, *Efficient implementation of essentially nonoscillatory shock-capturing schemes*, J. Comput. Phys., 77(2):439–471, 1988.

[13] C.-W. Shu and S. Osher, *Efficient implementation of essentially nonoscillatory shock-capturing schemes. II*, J. Comput. Phys., 83(1):32–78, 1989.

[14] S. Osher and C.-W. Shu, *High order essentially non-oscillatory schemes for Hamilton-Jacobi equations*, SIAM Journal on Numerical Analysis, v28 (1991), pp.907-922.

[15] M. Sussman, E. Fatemi, P. Smereka, and S. Osher, *An Improved Level Set Method for Incompressible Two-phase Flows*, Computers and Fluids, vol. 27, Nos 5-6 (1998), pp. 663-680.

[16] M. Sussman and M. Y. Hussaini, *A discontinuous spectral element method for the l evel set equation*, J. Sci. Comput., 19 (2003), no. 1-3, 479–500.

[17] M. Sussman, P. Smereka and S. Osher, *A Levelset Approach for computing solutions to incompressible two-phase flow*, J. Comp. Phys., vol. 114, (1994), pp. 146-159.

[18] Zhang, Y.-T. and Shu, C.-W., *High-order WENO schemes for Hamilton-Jacobi equations on triangular meshes*, SIAM J. Sci. Comput., vol. 24(3), (2002), pp. 1005–1030.

[19] S. O. Unversi and G. Tryggvason, *A front-tracking method for viscous, incompressible, multifluid flows*, J. Comp. Phys., vol 100, (1992), pp. 25-37.