A DISCONTINUOUS GALERKIN FINITE ELEMENT METHOD FOR HAMILTON-JACOBI EQUATIONS*

CHANGQING HU† AND CHI-WANG SHU‡

Abstract. In this paper, we present a discontinuous Galerkin finite element method for solving the nonlinear Hamilton-Jacobi equations. This method is based on the Runge-Kutta discontinuous Galerkin finite element method for solving conservation laws. The method has the flexibility of treating complicated geometry by using arbitrary triangulation, can achieve high order accuracy with a local, compact stencil, and are suited for efficient parallel implementation. One and two dimensional numerical examples are given to illustrate the capability of the method. At least k-th order of accuracy is observed for smooth problems when k-th degree polynomials are used, and derivative singularities are resolved well without oscillations even without limiters.

Key words. Hamilton-Jacobi Equations, discontinuous Galerkin, high order accuracy.

AMS subject classifications. 65M60, 70H20

1. Introduction. In this paper we consider the numerical solutions of Hamilton-Jacobi (HJ) equations

$$\varphi_t + H(\varphi_{x_1}, ..., \varphi_{x_d}) = 0, \qquad \varphi(x, 0) = \varphi^0(x). \tag{1.1}$$

As is well known, the solutions to (1.1) are Lipschitz continuous but may have discontinuous derivatives, regardless of the smoothness of the initial condition $\varphi^0(x)$. The non-uniqueness of such solutions also necessitates the definition of viscosity solutions, to single out a unique, practically relevant solution. See Crandall and Lions [12].

A widely applied class of numerical schemes for (1.1) is the class of finite difference schemes. In [13], Crandall and Lions proved the convergence of monotone finite difference schemes to the viscosity solutions of (1.1). Unfortunately, monotone schemes are at most first order accurate, measured by local truncation errors in smooth regions of the solution. In Osher and Sethian [23] and Osher and Shu [24], a class of high order essentially non-oscillatory (ENO) schemes were introduced, which were adapted from the methodologies for hyperbolic conservation laws [15], [26], [27]. The numerical results in [23] and [24] indicate convergence to the viscosity solutions of (1.1) with high order accuracy in smooth regions, and with sharp resolution of the discontinuous derivatives. Recently, weighted ENO (WENO) schemes for conservation laws [21], [18] have also been adapted to the Hamilton-Jacobi equations (1.1), [16].

Finite difference methods require a structured mesh, hence are difficult to apply to complicated geometry or for adaptive mesh refinements. Finite volume schemes, which are based on arbitrary triangulation, are thus attractive in such cases. First order monotone type finite volume schemes and their second extensions were studied in [2]. A second order ENO type finite volume scheme is developed in [20]. However,

*Research was supported in part by ARO grant DAAG55-97-1-0318, NSF grants DMS-9500814 and DMS-9804985, ECS-9627849 and INT-9601084, NASA Langley grant NAG-1-1145 and Contract NAS1-12070 while the second author was in residence at ICASE, NASA Langley Research Center, Hampton, VA 23681-0001, and AFOSR grant F49620-96-1-0150.

†Division of Applied Mathematics, Brown University, Providence, Rhode Island 02912 (hu@cfm.brown.edu).

‡Division of Applied Mathematics, Brown University, Providence, Rhode Island 02912 (shu@cfm.brown.edu).

higher order finite volume schemes face the problem of reconstruction on arbitrary triangulation, which is quite complicated. See, e.g., [1]. More recently, a continuous finite element method for solving (1.1) is proposed in [3].

The Runge-Kutta discontinuous Galerkin method [6, 7, 8, 9, 10] is a method devised to numerically solve the conservation law (CL):

$$u_t + f_1(u)_{x_1} + \cdots + f_d(u)_{x_d} = 0, \qquad u(x,0) = u^0(x). \tag{1.2}$$

The method has the following attractive properties:

- It can be designed for any order of accuracy in space and time. In fact, p-version or spectral element type version can be designed [22];
- It is an explicit method, thus efficient for solving the hyperbolic problem (1.2). No global linear or nonlinear systems must be solved;
- It combines the flexibility of finite element methods in the easy handling of complicated geometry, with the high resolution property for discontinuous solutions of finite difference and finite volume methods through monotone fluxes or approximate Riemann solvers applied at the element interfaces and limiters;
- It has nice stability properties: a local cell entropy inequality for the square entropy can be proven [17] for general triangulation for any scalar nonlinear conservation laws (1.2) in any spatial dimensions and for any order of accuracy, without the need of nonlinear limiters. This implies nonlinear L^2 stability and entropy consistency even for discontinuous solutions;
- The method is highly compact: the evolution of the information in any element depends only on the information of itself and its immediate neighbors, regardless of the order of accuracy. This is in contrast with high order finite volume schemes which must use wide stencils for the high order reconstruction. This compactness is responsible for the efficient parallel implementation of the method, see, e.g. [4].

For details of the Runge-Kutta discontinuous Galerkin method, see the references listed above and also the review paper of Cockburn [5]. This method can also be generalized to solve problems containing higher derivatives, [11], which is important for some of our applications in Sect. 4.

It is well known that the Hamilton-Jacobi equation (1.1) is closely related to the conservation law (1.2), in fact in one space dimension $d = 1$ they are equivalent if one takes $u = \varphi_x$. It is thus not surprising that many successful numerical methods for the Hamilton-Jacobi equation (1.1) are adapted from those for the conservation law (1.2). The high order finite difference ENO methods in [23] and [24] are such examples.

In this paper we adapt the Runge-Kutta discontinuous Galerkin method to solve the Hamilton-Jacobi equation (1.1). In Sect. 2 the algorithm in one space dimension is described and discussed. In Sect. 3 the algorithm for two space dimensions is developed. Numerical examples of one and two spatial dimensions are presented in Sect. 4. At least k-th order of accuracy is observed for smooth problems when k-th degree polynomials are used, and derivative singularities are resolved well without oscillations even without limiters. Concluding remarks are given in Sect. 5.

2. One Dimensional Case. In one space dimension (1.1) becomes

$$\varphi_t + H(\varphi_x) = 0, \qquad \varphi(x,0) = \varphi^0(x). \tag{2.1}$$

This is a relatively easy case because (2.1) is equivalent to the conservation law

$$u_t + H(u)_x = 0, \qquad u(x,0) = u^0(x) \tag{2.2}$$

if we identify $u = \varphi_x$.

Assuming (2.1) is solved in the interval $a \le x \le b$ and it is divided into the following cells:

$$a = x_{\frac{1}{2}} < x_{\frac{3}{2}} < \cdots < x_{N+\frac{1}{2}} = b, \tag{2.3}$$

we denote

$$I_j = (x_{j-\frac{1}{2}}, x_{j+\frac{1}{2}}),\ x_j = \frac{1}{2}\left(x_{j-\frac{1}{2}} + x_{j+\frac{1}{2}}\right),\ h_j = x_{j+\frac{1}{2}} - x_{j-\frac{1}{2}},\ h = \max_j h_j, \tag{2.4}$$

and define the following approximation space

$$V_h^k = \left\{ v : v|_{I_j} \in P^k(I_j), j = 1, ..., N \right\}. \tag{2.5}$$

Here $P^k(I_j)$ is the set of all polynomials of degree at most k on the cell I_j.

A k-th order discontinuous Galerkin scheme for the one dimensional Hamilton-Jacobi equation (2.1) can then be defined as follows: find $\varphi \in V_h^k$, such that

$$\int_{I_j} \varphi_{xt} v \, dx - \int_{I_j} H(\varphi_x) v_x \, dx + \hat{H}_{j+\frac{1}{2}} v^-_{j+\frac{1}{2}} - \hat{H}_{j-\frac{1}{2}} v^+_{j-\frac{1}{2}} \;=\; 0, \qquad j = 1, ..., N, \tag{2.6}$$

for all $v \in V_h^{k-1}$. Here

$$\hat{H}_{j+\frac{1}{2}} = \hat{H}\left((\varphi_x)^-_{j+\frac{1}{2}}, (\varphi_x)^+_{j+\frac{1}{2}} \right) \tag{2.7}$$

is a monotone flux, i.e. $\hat{H}$ is non-decreasing in the first argument and non-increasing in the second, symbolically $\hat{H}(\uparrow, \downarrow)$, is Lipschitz continuous in both arguments, and is consistent, i.e. $\hat{H}(u,u) = H(u)$. We will mainly use the simple (local) Lax-Friedrichs flux

$$\hat{H}(u^-, u^+) = H\left(\frac{u^- + u^+}{2}\right) - \frac{1}{2}\alpha\,(u^+ - u^-) \tag{2.8}$$

where $\alpha = \max_u |H'(u)|$ with the maximum taken over the range covered by u^- and u^+. For other monotone fluxes, e.g. the Godunov flux, see [24]. Notice that the method described above is exactly the discontinuous Galerkin method for the conservation law equation (2.2) satisfied by the derivative $u = \varphi_x$. This only determines φ for each element up to a constant, since it is only a scheme for φ_x. The missing constant can be obtained in one of the following two ways:

1. By requiring that

$$\int_{I_j} (\varphi_t + H(\varphi_x))\, v \, dx \;=\; 0, \qquad j = 1, ..., N, \tag{2.9}$$

for all $v \in V_h^0$, that is,

$$\int_{I_j} (\varphi_t + H(\varphi_x))\, dx \;=\; 0, \qquad j = 1, ..., N. \tag{2.10}$$

2. By using (2.10) to update only one (or a few) elements, e.g., the left-most element I_1, then use

$$\varphi(x_j, t) = \varphi(x_1, t) + \int_{x_1}^{x_j} \varphi_x(x, t)\, dx \tag{2.11}$$

to determine the missing constant for the cell I_j.

We remark that, in the second approach, the recovered values of φ are dependent upon the choice of the starting point x_1. However this difference is on the level of truncation errors and does not affect the order of accuracy. See Examples 4.4 and 4.6 in Sect. 4. Both approaches are used in our numerical experiments. They perform similarly for smooth problems, with the first approach giving slightly better results. However, it is our numerical experience that, when there are singularities in the derivatives, the first approach will often produce dents and bumps when the integral path in time passes through the singularities at some earlier time. The philosophy of using the second approach is that one could update only a few elements whose time integral paths do not cross derivative singularities. The numerical results shown in Sect. 4 are obtained with the second approach.

About the stability of the method proposed above, we can quote the following result of Jiang and Shu [17]. Here we assume compact support or periodic boundary condition for φ.

Lemma 2.1. [17]. The following L^2 stability result for the derivative φ_x holds for the discontinuous Galerkin method defined above, of any order of accuracy k applied to any nonlinear Hamilton-Jacobi equation (2.1):

$$\frac{d}{dt} \int_a^b \varphi_x^2\, dx \le 0. \tag{2.12}$$

□

For a finite interval $[a, b]$, (2.12) trivially implies TVB (total variation bounded) property for the numerical solution φ:

$$TV(\varphi) = \int_a^b |\varphi_x|\, dx \le \sqrt{b-a}\sqrt{\int_a^b \left(\frac{d}{dx}\varphi^0(x)\right)^2 dx}. \tag{2.13}$$

This is a rather strong stability result, considering that it applies even if the derivative of the solution φ_x develops discontinuities, no limiter has been added to the numerical scheme, and the scheme can be of arbitrary high order in accuracy. It also implies convergence of at least a subsequence of the numerical solution φ when $h \to 0$. However, this stability result is not strong enough to imply that the limit solution is the viscosity solution of (2.1).

Up to now we have only described the spatial discretization and have left the time variable t continuous (the method of lines approach). Time discretization is by the TVD (total variation diminishing) high order Runge-Kutta methods developed in [26], see also [14]. The second and third order versions used in this paper are as follows: for solving the method of lines ODE

$$\varphi_t = L(\varphi), \tag{2.14}$$

the second order TVD Runge-Kutta method is given by

$$\begin{aligned}\varphi^{(1)} &= \varphi^n + \Delta t L(\varphi^n)\\ \varphi^{n+1} &= \frac{1}{2}\varphi^n + \frac{1}{2}\varphi^{(1)} + \frac{1}{2}\Delta t L(\varphi^{(1)}),\end{aligned} \tag{2.15}$$

and the third order TVD Runge-Kutta method is given by

$$\begin{aligned}\varphi^{(1)} &= \varphi^n + \Delta t L(\varphi^n)\\ \varphi^{(2)} &= \frac{3}{4}\varphi^n + \frac{1}{4}\varphi^{(1)} + \frac{1}{4}\Delta t L(\varphi^{(1)})\\ \varphi^{n+1} &= \frac{1}{3}\varphi^n + \frac{2}{3}\varphi^{(2)} + \frac{2}{3}\Delta t L(\varphi^{(2)}).\end{aligned} \tag{2.16}$$

These Runge-Kutta methods will also be used for the two dimensional case discussed in the next section.

3. Two Dimensional Case. We consider in this section the case of two spatial dimensions. The algorithm in more spatial dimensions is similar. This time, the scalar Hamilton-Jacobi equation

$$\varphi_t + H(\varphi_x, \varphi_y) = 0, \qquad \varphi(x,y,0) = \varphi^0(x,y) \tag{3.1}$$

is in some sense equivalent to the following conservation law *system*

$$u_t + H(u,v)_x = 0, \qquad v_t + H(u,v)_y = 0, \qquad (u,v)(x,y,0) = (u,v)^0(x,y). \tag{3.2}$$

if we identify

$$(u,v) = (\varphi_x, \varphi_y). \tag{3.3}$$

For example, a vanishing viscosity solution of (3.1) corresponds, via (3.3), to a vanishing viscosity solution of (3.2), and vice versa [19]. However, (3.2) is not a strictly hyperbolic system, which may cause problems in its numerical solution if we treat u and v as independent variables. Instead, we would like to still use φ as our solution variable (a polynomial) and take its derivatives as u and v.

Assuming we are solving (3.1) in the domain Ω, which has a triangulation $\mathcal{T}_h$ consisting of triangles or general polygons of maximum size (diameter) h, with the following approximation space

$$V_h^k = \left\{ v : v|_K \in P^k(K),\ \forall K \in \mathcal{T}_h \right\}, \tag{3.4}$$

where $P^k(K)$ is again the set of all polynomials of degree at most k on the cell K. We propose a discontinuous Galerkin method for (3.1) as follows: find $\varphi \in V_h^k$, such that

$$\int_K \varphi_{xt} v \, dxdy - \int_K H(\varphi_x, \varphi_y) v_x \, dxdy + \sum_{e \in \partial K} \int_e \hat{H}_{1,e,K} \, v \, d\Gamma = 0 \tag{3.5}$$

and

$$\int_K \varphi_{yt} v \, dxdy - \int_K H(\varphi_x, \varphi_y) v_y \, dxdy + \sum_{e \in \partial K} \int_e \hat{H}_{2,e,K} \, v \, d\Gamma = 0 \tag{3.6}$$

for all $v \in V_h^{k-1}$ and all $K \in \mathcal{T}_h$, *in a least square sense*. Here the numerical flux is

$$\hat{H}_{i,e,K} = \hat{H}_{i,e,K}\left((\nabla\varphi)^{int(K)}, (\nabla\varphi)^{ext(K)}\right), \qquad i = 1,2 \tag{3.7}$$

where the superscript $int(K)$ implies that the value is taken from within the element K, and the superscript $ext(K)$ implies that the value is taken from outside the element K and within the neighboring element K' sharing the edge e with K. The flux (3.7) satisfies the following properties:

1. $\hat{H}_{i,e,K}$ is Lipschitz continuous with respect to all its arguments;
2. Consistency:
$$\hat{H}_{i,e,K}\left(\nabla\varphi, \nabla\varphi\right) = H(\nabla\varphi)\, n_i,$$
where $n = (n_1, n_2)$ is the unit outward normal to the edge e of the element K;
3. Conservation:
$$\hat{H}_{i,e,K}\left((\nabla\varphi)^{int(K)}, (\nabla\varphi)^{ext(K)}\right) = -\hat{H}_{i,e,K'}\left((\nabla\varphi)^{int(K')}, (\nabla\varphi)^{ext(K')}\right).$$
where $K \cap K' = e$.

We will again mainly use the simple (local) Lax-Friedrichs flux

$$\hat{H}_{1,e,K}((u,v)^-, (u,v)^+) = H\left(\frac{u^- + u^+}{2}, \frac{v^- + v^+}{2}\right) n_1 - \frac{1}{2}\alpha(u^+ - u^-) \tag{3.8}$$

and

$$\hat{H}_{2,e,K}((u,v)^-, (u,v)^+) = H\left(\frac{u^- + u^+}{2}, \frac{v^- + v^+}{2}\right) n_2 - \frac{1}{2}\beta(v^+ - v^-) \tag{3.9}$$

where $\alpha = \max_{u,v} |\frac{\partial H(u,v)}{\partial u}|$ and $\beta = \max_{u,v} |\frac{\partial H(u,v)}{\partial v}|$, with the maximum being taken over the relevant (local) range.

Notice that (3.5)-(3.6) are exactly the discontinuous Galerkin method for the conservation law system (3.2) satisfied by the derivatives $(u,v) = \nabla\varphi$. For a rectangular mesh and for $k = 1$ this recovers the (local) Lax-Friedrichs monotone scheme [24] for (3.1) if we identify $u_{ij} = \frac{\varphi_{i+1,j} - \varphi_{i-1,j}}{2\Delta x}$ and $v_{ij} = \frac{\varphi_{i,j+1} - \varphi_{i,j-1}}{2\Delta y}$.

Of course, (3.5)-(3.6) have more equations than the number of degrees of freedoms (an over-determined system) for $k > 1$, thus a least square solution is needed. In practice, the least square procedure is performed as follows: we first evolve (3.2) for one time step (one inner stage for high order Runge-Kutta methods), using the discontinuous Galerkin method (3.5)-(3.6) with $u = \varphi_x$ and $v = \varphi_y$; then φ at the next time level (stage) is obtained (up to a constant) by least square:

$$||(\varphi_x - u)^2 + (\varphi_y - v)^2||_{L^1(K)} = \min_{\psi \in P^k(K)} ||(\psi_x - u)^2 + (\psi_y - v)^2||_{L^1(K)}.$$

This determines φ for each element up to a constant, since it is only a scheme for $\nabla\varphi$. The missing constant can again be obtained in one of the following two ways:

1. By requiring that
$$\int_K (\varphi_t + H(\varphi_x, \varphi_y))\, v\, dxdy \;=\; 0, \tag{3.10}$$

for all $v \in V_h^0$ and for all $K \in \mathcal{T}_h$, that is,

$$\int_K (\varphi_t + H(\varphi_x, \varphi_y))\, dxdy = 0, \qquad \forall K \in \mathcal{T}_h\,; \tag{3.11}$$

2. By using (3.11) to update only one (or a few) elements, e.g., the corner element(s), then use

$$\varphi(B,t) = \varphi(A,t) + \int_A^B (\varphi_x\, dx + \varphi_y\, dy) \tag{3.12}$$

to determine the missing constant. The path should be taken to avoid crossing a derivative discontinuity, if possible.

We remark again that, in the second approach, the recovered values of φ are dependent upon the choice of the starting point A as well as the integration path. However this difference is on the level of truncation errors and does not affect the order of accuracy. See Examples 4.4 and 4.6 in Sect. 4. It is important here that φ is a single function and u and v are just its x and y derivatives. Otherwise the second approach would be questionable in effectively recovering φ.

In Sect. 4 we will only show numerical results obtained with the second approach, see the remarks in the previous section for the one dimensional case.

We remark that the procedure discussed above is easily implemented in any triangulation, e.g., for both rectangles and triangles.

4. Numerical Examples.

We now present some 1D and 2D numerical results.

Example 4.1. One dimensional Burgers' equation:

$$\begin{cases} \varphi_t + \frac{(\varphi_x+1)^2}{2} = 0, & -1 < x < 1 \\ \varphi(x,0) = -\cos(\pi x) \end{cases} \tag{4.1}$$

with periodic boundary conditions.

The local Lax-Friedrichs flux (2.8) is used. At $t = 0.5/\pi^2$, the solution is still smooth. We list the errors and the numerical orders of accuracy in Table 4.1 We observe that, except for the P^1 case which seems to be only first order, P^k for $k > 1$ seems to provide close to $(k+1)$-th order accuracy. The meshes used are all uniform, and errors are computed at the middle point of each interval.

To investigate the accuracy problem further, we use non-uniform meshes obtained by randomly shifting the cell boundaries in a uniform mesh in the range $[-0.1h, 0.1h]$. In order to avoid possible superconvergence at cell centers, we also give the "real" L^2 error (computed by a 6-point Gaussian quadrature in each cell). The results are shown in Table 4.2.

At $t = 3.5/\pi^2$, the solution has developed a discontinuous derivative. In Fig. 4.1, we show the sharp corner-like numerical solution with 41 elements obtained with P^k for $k = 1, 2, 3, 4$ with a uniform mesh. Here and below, the solid line is the exact solution, the circles are numerical solutions (only one point per element is drawn).

Example 4.2. One dimensional equation with a non-convex flux:

$$\begin{cases} \varphi_t - \cos(\varphi_x + 1) = 0, & -1 < x < 1 \\ \varphi(x,0) = -\cos(\pi x) \end{cases} \tag{4.2}$$

TABLE 4.1
Accuracy for 1D Burgers equation (uniform mesh), $t = 0.5/\pi^2$.

	P^1		P^2		P^3		P^4	
N	L^1 error	order	L^1 error	order	L^1 error	order	L^1 error	order
10	0.17E+00	—	0.14E-02	—	0.21E-03	—	0.57E-05	—
20	0.78E-01	1.12	0.18E-03	2.92	0.13E-04	3.94	0.73E-06	2.97
40	0.35E-01	1.16	0.24E-04	2.97	0.75E-06	4.17	0.32E-07	4.52
80	0.16E-01	1.12	0.28E-05	3.08	0.43E-07	4.12	0.12E-08	4.79
160	0.76E-02	1.02	0.31E-06	3.19	0.25E-08	4.10	0.48E-10	4.59

	P^1		P^2		P^3		P^4	
N	L^∞ error	order	L^∞ error	order	L^∞ error	order	L^∞ error	order
10	0.29E+00	—	0.24E-02	—	0.69E-03	—	0.13E-04	—
20	0.13E+00	1.13	0.33E-03	2.88	0.61E-04	3.51	0.16E-05	2.99
40	0.58E-01	1.15	0.37E-04	3.15	0.58E-05	3.39	0.13E-06	3.64
80	0.27E-01	1.11	0.48E-05	2.97	0.38E-06	3.93	0.59E-08	4.44
160	0.13E-01	1.07	0.59E-06	3.00	0.23E-07	4.07	0.25E-09	4.57

with periodic boundary conditions.

The local Lax-Friedrichs flux (2.8) and uniform meshes are used. At $t = 0.5/\pi^2$, the solution is still smooth. The accuracy of the numerical solution is listed in Table 4.3. We observe similar accuracy as in the previous example.

At $t = 1.5/\pi^2$, the solution has developed corner-like discontinuity in the derivative. The numerical result with 41 elements is shown in Fig. 4.2.

Example 4.3. The one dimensional Riemann problem with a non-convex flux:

$$\begin{cases} \varphi_t + \frac{1}{4}(\varphi_x^2 - 1)(\varphi_x^2 - 4) = 0, & -1 < x < 1 \\ \varphi(x,0) = -2|x| \end{cases} \tag{4.3}$$

For this test problem, the discontinuous Galerkin method, as described in Sect. 2 and applied in the previous two examples, fails to "open up" the initial single discontinuity in the derivative sufficiently to generate the correct entropy solution. We have found out that a nonlinear total variation bounded limiting, described in detail in [7] for this one dimensional case, is needed for convergence towards the entropy solution. This and the 2D Riemann problem in Example 4.7 below are the only two examples in this paper in which we use the nonlinear limiting. We remark that for the finite difference schemes, such nonlinear limiting or the adaptive stencil in ENO is needed in most cases in order to enforce stability and to obtain non-oscillatory results.

Numerical results at $t = 1$ with 81 elements, using the local Lax-Friedrichs flux (2.8), is shown in Fig. 4.3. The results of using the Godunov flux is shown in Fig. 4.4. We can see that while for P^1, the results of using two different monotone fluxes are significantly different in resolution, this difference is greatly reduced for higher order of accuracy. In most of the high order cases, the simple local Lax-Friedrichs flux (2.8) should be good enough.

TABLE 4.2
Accuracy for 1D Burgers equation (non-uniform mesh), $t = 0.5/\pi^2$.

	P^1		P^2		P^3		P^4	
N	L^2 error	order	L^2 error	order	L^2 error	order	L^2 error	order
10	0.74E+00	—	0.34E-02	—	0.32E-03	—	0.53E-04	—
20	0.34E+00	1.11	0.51E-03	2.76	0.24E-04	3.72	0.20E-05	4.71
40	0.15E+00	1.19	0.65E-04	2.96	0.17E-05	3.82	0.71E-07	4.84
80	0.67E-01	1.17	0.90E-05	2.86	0.13E-06	3.72	0.20E-08	5.15
160	0.31E-01	1.13	0.11E-05	3.02	0.81E-08	4.02	0.74E-10	4.76

	P^1		P^2		P^3		P^4	
N	L^1 error	order	L^1 error	order	L^1 error	order	L^1 error	order
10	0.53E+00	—	0.17E-02	—	0.23E-03	—	0.30E-05	—
20	0.24E+00	1.13	0.21E-03	3.05	0.14E-04	4.01	0.40E-06	2.89
40	0.11E+00	1.19	0.26E-04	2.99	0.78E-06	4.20	0.16E-07	4.65
80	0.47E-01	1.17	0.37E-05	2.82	0.47E-07	4.05	0.61E-09	4.70
160	0.21E-01	1.13	0.41E-06	3.16	0.27E-08	4.15	0.26E-10	4.56

	P^1		P^2		P^3		P^4	
N	L^∞ error	order	L^∞ error	order	L^∞ error	order	L^∞ error	order
10	0.62E+00	—	0.36E-02	—	0.69E-03	—	0.11E-04	—
20	0.29E+00	1.11	0.47E-03	2.94	0.61E-04	3.52	0.16E-05	2.81
40	0.13E+00	1.16	0.67E-04	2.80	0.47E-05	3.70	0.13E-06	3.64
80	0.58E-01	1.14	0.17E-04	2.01	0.62E-06	2.91	0.59E-08	4.45
160	0.27E-01	1.11	0.19E-05	3.11	0.31E-07	4.32	0.33E-09	4.17

TABLE 4.3
Accuracy for 1D non-convex, $H(u) = -\cos(u+1)$, $t = 0.5/\pi^2$.

	P^1		P^2		P^3		P^4	
N	L^1 error	order	L^1 error	order	L^1 error	order	L^1 error	order
10	0.84E-01	—	0.10E-02	—	0.34E-03	—	0.24E-04	—
20	0.36E-01	1.23	0.15E-03	2.75	0.30E-04	3.49	0.13E-05	4.28
40	0.15E-01	1.26	0.21E-04	2.84	0.15E-05	4.33	0.59E-07	4.42
80	0.68E-02	1.14	0.27E-05	2.97	0.94E-07	4.00	0.21E-08	4.78

	P^1		P^2		P^3		P^4	
N	L^∞ error	order	L^∞ error	order	L^∞ error	order	L^∞ error	order
10	0.18E+00	—	0.15E-02	—	0.11E-02	—	0.99E-04	—
20	0.73E-01	1.31	0.27E-03	2.43	0.22E-03	2.35	0.13E-04	2.95
40	0.31E-01	1.24	0.47E-04	2.54	0.18E-04	3.63	0.59E-06	4.44
80	0.14E-01	1.16	0.85E-05	2.47	0.14E-05	3.75	0.26E-07	4.49

Example 4.4. Two dimensional Burgers' equation:

$$\begin{cases} \varphi_t + \frac{(\varphi_x+\varphi_y+1)^2}{2} = 0, & -2 < x < 2,\ -2 < y < 2 \\ \varphi(x,y,0) = -\cos\left(\frac{\pi(x+y)}{2}\right) \end{cases} \tag{4.4}$$

with periodic boundary conditions.

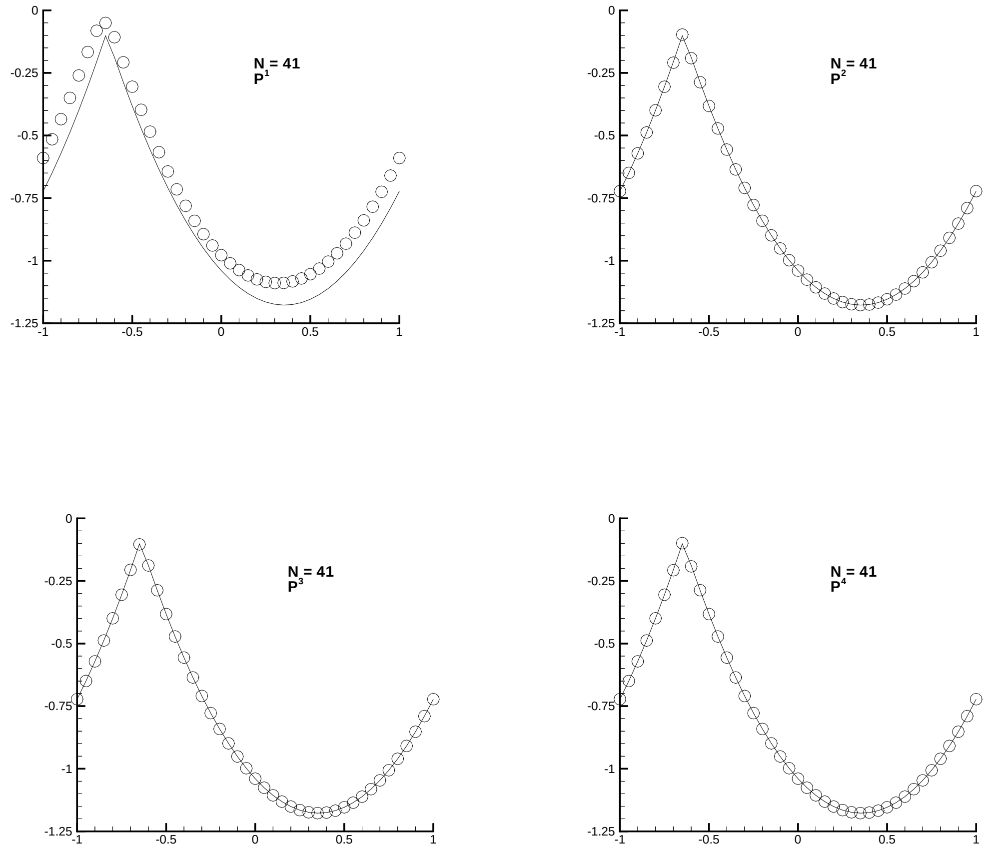

FIG. 4.1. *One-dimensional Burgers' equation,* $t = 3.5/\pi^2$.

We first use uniform rectangular meshes and the local Lax-Friedrichs flux (3.8)-(3.9). At $t = 0.5/\pi^2$, the solution is still smooth. The errors (computed at the center of the cells) and orders of accuracy are listed in Table 4.4. It seems that only k-th order of accuracy is achieved when φ is a piecewise polynomial of degree k.

Next, as in the one dimensional case, we use non-uniform rectangular meshes obtained from the tensor product of one dimensional nonuniform meshes (the meshes in two directions are independent). Again, we give the "real" L^2-errors computed by a 6×6 point Gaussian quadrature as well. The results are shown in Table 4.5.

The results in the previous two tables are obtained by updating the element at the left-lower corner with time, and then taking an integration path consisting of line segments starting from the corner and parallel to the x-axis first, then vertically to the point. To further address the issue of the dependency of the computed values of the solution φ on the integration path and starting point, we use another path which starts vertically, then parallelly with the x-axis to reach the point. In Table 4.6, we list the *difference* of two recovered solutions φ from these two different integration paths, for the non-uniform mesh cases. We can see that these differences are at the levels of local truncation errors and decay in the same order as the errors. Thus the

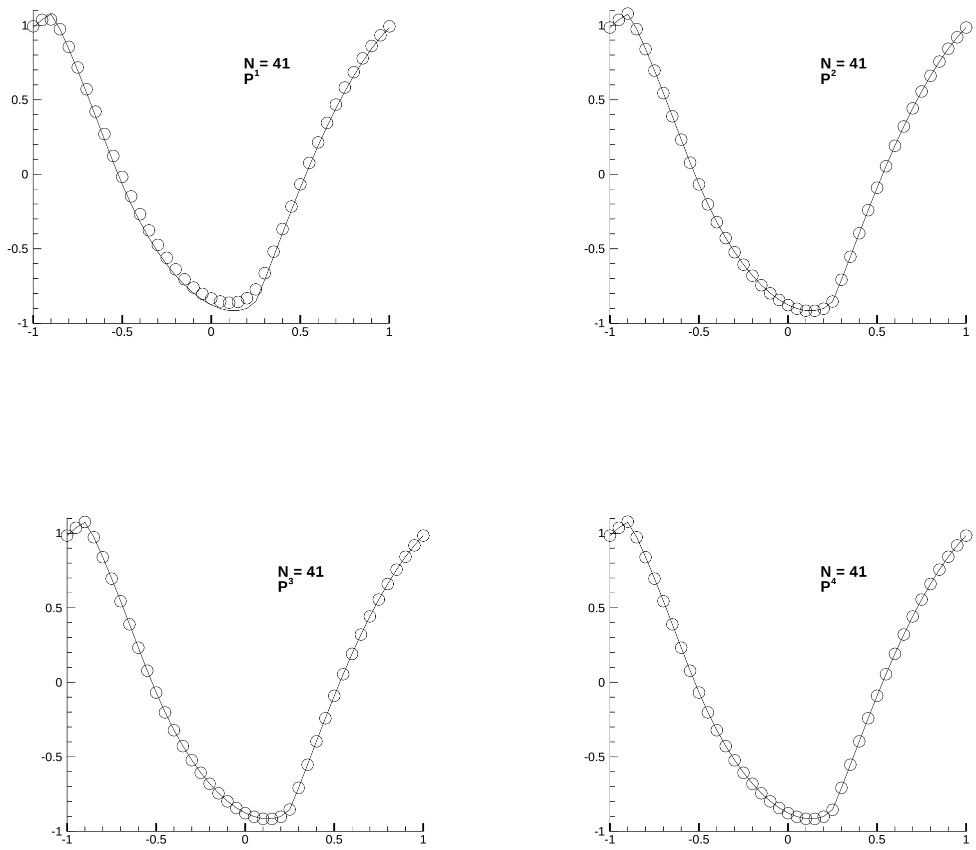

FIG. 4.2. *One dimension non-convex,* $H(u) = -\cos(u+1)$, $t = 1.5/\pi^2$.

choice of integration path in recovering φ does not affect accuracy.

At $t = 1.5/\pi^2$, the solution has discontinuous derivatives. Fig. 4.5 is the graph of the numerical solution with 40×40 elements (uniform mesh).

Finally we use triangle based triangulation, the mesh with $h = \frac{1}{4}$ is shown in Fig. 4.6. The accuracy at $t = 0.5/\pi^2$ is shown in Table 4.7. Similar accuracy pattern is observed as in the rectangular case. The result at $t = 1.5/\pi^2$, when the derivative is discontinuous, is shown in Fig. 4.7.

Example 4.5. Two dimensional equation:

$$\left\{ \begin{array}{ll} \varphi_t - \cos(\varphi_x + \varphi_y + 1) = 0, & -2 < x < 2,\ -2 < y < 2 \\ \varphi(x, y, 0) = -\cos\left(\frac{\pi(x+y)}{2}\right) & \end{array} \right. \tag{4.5}$$

with periodic boundary conditions.

For this example we use uniform rectangular meshes. The local Lax-Friedrichs flux (3.8)-(3.9) is used. The solution is smooth at $t = 0.5/\pi^2$. The accuracy of the numerical solution is shown in Table 4.8.

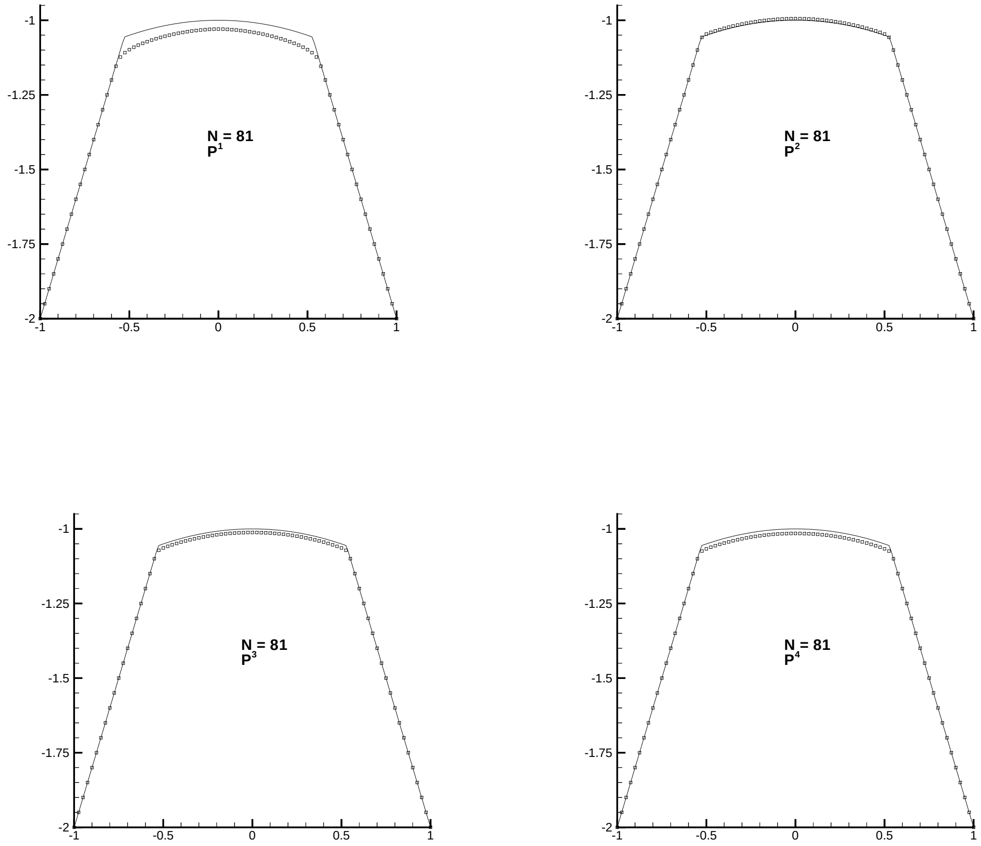

FIG. 4.3. *One dimension Riemann problem, local Lax-Friedrichs flux,* $H(u) = \frac{1}{4}(u^2-1)(u^2-4), t = 1$.

The solution has developed a discontinuous derivative at $t = 1.5/\pi^2$. Results with 40×40 elements are shown in Fig. 4.8.

Example 4.6. The level set equation in a domain with a hole:

$$\begin{cases} \varphi_t + \mathrm{sign}(\varphi_0)(\sqrt{\varphi_x^2 + \varphi_y^2} - 1) = 0, & \frac{1}{2} < \sqrt{x^2+y^2} < 1 \\ \varphi(x, y, 0) = \varphi_0(x, y) \end{cases} \tag{4.6}$$

This problem is introduced in [28]. The solution φ to (4.6) has the same zero level set as φ_0, and the steady state solution is the distance function to that zero level curve. We use this problem to test the effects using various integration paths (3.12) when there is a hole in the region. Notice that the exact steady state solution is the distance function to the inner boundary of domain when boundary condition is adequately prescribed. We take φ_0 equal to this distance function, and compute the time dependent problem to reach a steady state solution, using the exact steady state solution for the boundary conditions. Four symmetric elements near the outer

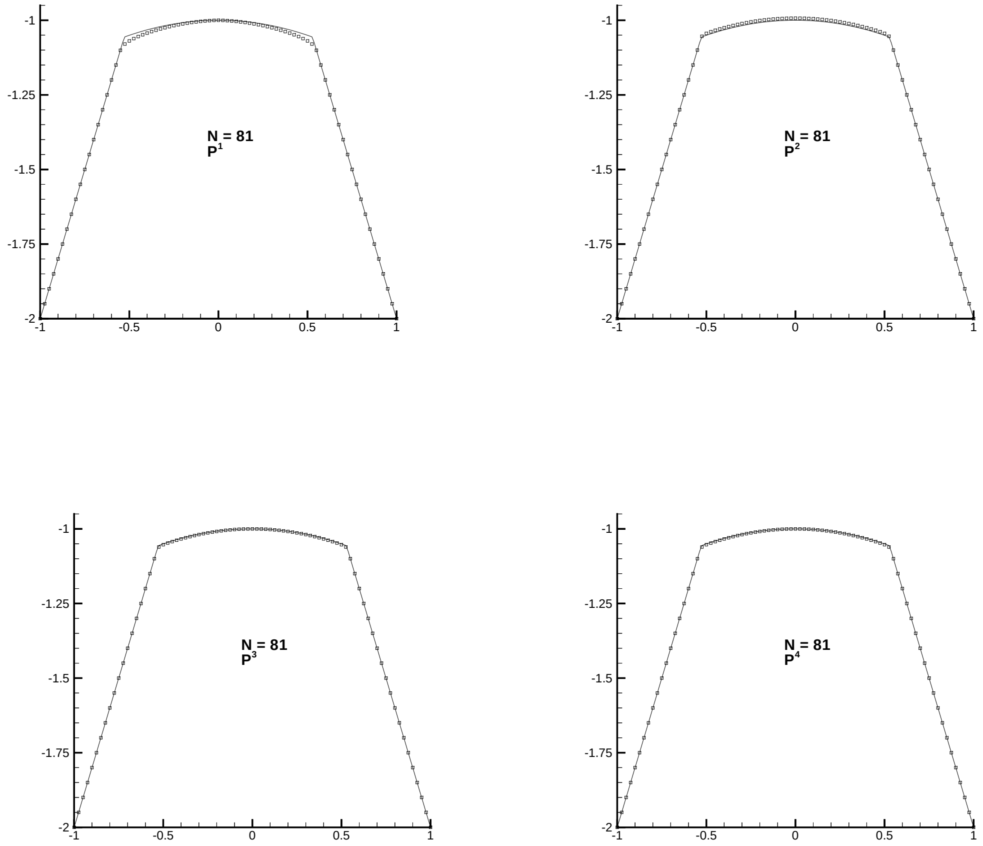

FIG. 4.4. *One dimension Riemann problem, Godunov flux,* $H(u) = \frac{1}{4}(u^2-1)(u^2-4), t = 1$.

boundary are updated by the exact steady state solution, all other elements are recovered from (3.12) by the shortest path to the nearest one of above four elements. The results are shown in Table 4.9. Also shown in Table 4.9 is the error (difference) between the numerical solution φ thus recovered, and the value of φ after another integration along a circular path (starting and ending at the same point in (3.12)). We can see that the difference is small with the correct order of accuracy, further indicating that the dependency of the recovered solution φ on the integration path is on the order of the truncation errors even for such problems with holes. Finally, the mesh with 1432 triangles and the solution with 5608 triangles are shown in Fig. 4.9.

Example 4.7. Two dimensional Riemann problem:

$$\begin{cases} \varphi_t + \sin(\varphi_x + \varphi_y) = 0, & -1 < x < 1, -1 < y < 1 \\ \varphi(x, y, 0) = \pi(|y| - |x|) \end{cases} \tag{4.7}$$

For this example we use a uniform rectangular mesh with 40×40 elements. The local Lax-Friedrichs flux (3.8)-(3.9) is used. As was mentioned in Example 4.3, we

TABLE 4.4
Accuracy for 2D Burgers equation, uniform rectangular mesh, $t = 0.5/\pi^2$.

	P^1		P^2		P^3	
$N \times N$	L^1 error	order	L^1 error	order	L^1 error	order
10×10	8.09E-02	—	8.62E-03	—	3.19E-03	—
20×20	3.36E-02	1.268	1.72E-03	2.325	3.49E-04	3.192
40×40	1.48E-02	1.183	3.93E-04	2.130	6.64E-05	2.394
80×80	6.88E-03	1.105	9.74E-05	2.013	1.14E-05	2.542
160×160	3.31E-03	1.056	2.45E-05	1.991	1.68E-06	2.763

	P^1		P^2		P^3	
$N \times N$	L^∞ error	order	L^∞ error	order	L^∞ error	order
10×10	2.62E-01	—	3.56E-02	—	8.65E-03	—
20×20	1.14E-01	1.201	8.40E-03	2.083	1.16E-03	2.899
40×40	5.00E-02	1.189	2.02E-03	2.056	1.98E-04	2.551
80×80	2.39E-02	1.065	4.92E-04	2.038	3.13E-05	2.661
160×160	1.16E-02	1.043	1.21E-04	2.024	4.41E-06	2.827

TABLE 4.5
Accuracy for 2D Burgers equation, non-uniform rectangular mesh, $t = 0.5/\pi^2$.

	P^1		P^2		P^3	
$N \times N$	L^2 error	order	L^2 error	order	L^2 error	order
10×10	4.47E-01	—	6.28E-02	—	1.61E-02	—
20×20	1.83E-01	1.288	1.50E-02	2.066	2.06E-03	2.966
40×40	8.01E-02	1.192	3.63E-03	2.047	3.48E-04	2.565
80×80	3.82E-02	1.068	9.17E-04	1.985	6.03E-05	2.529
160×160	1.87E-02	1.031	2.34E-04	1.970	8.58E-06	2.813

	P^1		P^2		P^3	
$N \times N$	L^1 error	order	L^1 error	order	L^1 error	order
10×10	8.16E-02	—	9.16E-03	—	3.39E-03	—
20×20	3.41E-02	1.259	2.09E-03	2.132	4.12E-04	3.041
40×40	1.50E-02	1.185	5.21E-04	2.004	7.03E-05	2.551
80×80	7.16E-03	1.067	1.42E-04	1.875	1.24E-05	2.503
160×160	3.50E-03	1.033	3.85E-05	1.883	1.76E-06	2.817

	P^1		P^2		P^3	
$N \times N$	L^∞ error	order	L^∞ error	order	L^∞ error	order
10×10	2.83E-01	—	4.68E-02	—	1.00E-02	—
20×20	1.25E-01	1.179	1.23E-02	1.928	1.39E-03	2.847
40×40	5.74E-02	1.123	3.54E-03	1.797	2.29E-04	2.602
80×80	2.78E-02	1.046	1.15E-03	1.622	5.11E-05	2.164
160×160	1.42E-02	0.969	2.72E-04	2.080	7.16E-06	2.835

have found out that a nonlinear limiting [9], [10] is needed, for convergence towards an entropy solution. We show the numerical solution at $t = 1$ in Fig. 4.10.

TABLE 4.6
Differences of the solution φ recovered by two different integration paths, non-uniform mesh, Burgers equation.

	P^1		P^2		P^3	
$N \times N$	L^1 error	order	L^1 error	order	L^1 error	order
10×10	8.61E-03	—	2.90E-03	—	1.15E-03	—
20×20	4.64E-03	0.892	1.28E-03	1.180	2.44E-04	2.237
40×40	2.54E-03	0.869	4.12E-04	1.635	3.76E-05	2.698
80×80	1.81E-03	0.489	1.39E-04	1.568	6.71E-06	2.486
160×160	1.09E-03	0.732	3.66E-05	1.925	8.79E-07	2.932

TABLE 4.7
Accuracy for 2D Burgers equation, triangular mesh, $t = 0.5/\pi^2$.

	P^2				P^3			
h	L^1 error	order	L^∞ error	order	L^1 error	order	L^∞ error	order
1	5.48E-02	—	1.52E-01	—	1.17E-02	—	2.25E-02	—
1/2	1.35E-02	2.02	6.26E-02	1.28	1.35E-03	3.12	4.12E-03	2.45
1/4	2.94E-03	2.20	1.55E-02	2.01	1.45E-04	3.22	4.31E-04	3.26
1/8	6.68E-04	2.14	3.44E-03	2.17	1.71E-05	3.08	7.53E-05	2.52

TABLE 4.8
Accuracy, 2D, $H(u,v) = -\cos(u+v+1), t = 0.5/\pi^2$.

	P^1		P^2		P^3	
$N \times N$	L^1 error	order	L^1 error	order	L^1 error	order
10×10	6.47E-02	—	8.31E-03	—	1.35E-02	—
20×20	2.54E-02	1.349	1.93E-03	2.106	1.57E-03	3.104
40×40	1.05E-02	1.274	4.58E-04	2.075	2.39E-04	2.716
80×80	4.74E-03	1.147	1.13E-04	2.019	2.89E-05	3.048
160×160	2.23E-03	1.088	2.83E-05	1.997	4.38E-06	2.722

	P^1		P^2		P^3	
$N \times N$	L^∞ error	order	L^∞ error	order	L^∞ error	order
10×10	1.47E-01	—	1.88E-02	—	2.36E-02	—
20×20	6.75E-02	1.123	7.34E-03	1.357	3.44E-03	2.778
40×40	2.65E-02	1.349	1.83E-03	2.004	4.59E-04	2.906
80×80	1.18E-02	1.167	4.55E-04	2.008	5.78E-05	2.989
160×160	2.23E-03	1.088	1.13E-04	2.010	8.54E-06	2.759

TABLE 4.9
Errors for the level set equation, triangular mesh with P^2.

	Errors for the Solution				Errors by Integration Path			
N	L^1 error	order	L^∞ error	order	L^1 error	order	L^∞ error	order
403	1.02E-03	—	1.32E-03	—	1.61E-04	—	5.71E-04	—
1432	1.23E-04	3.05	2.73E-04	2.27	5.84E-05	1.46	1.68E-04	1.78
5608	1.71E-05	2.85	3.18E-05	3.10	9.32E-06	2.65	4.36E-05	1.95
22238	2.09E-06	3.03	5.01E-06	2.67	1.43E-06	2.70	6.63E-06	2.72

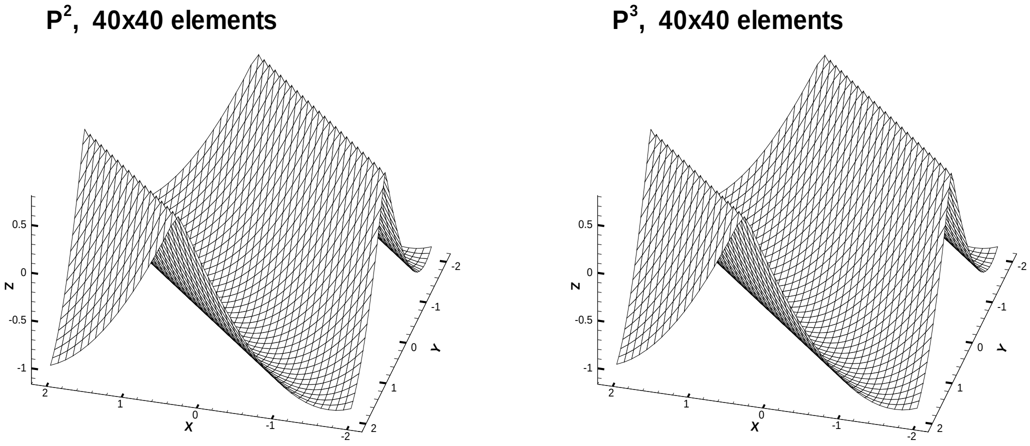

FIG. 4.5. *Two dimension Burgers' equation, rectangular mesh, t=1.5/π^2.*

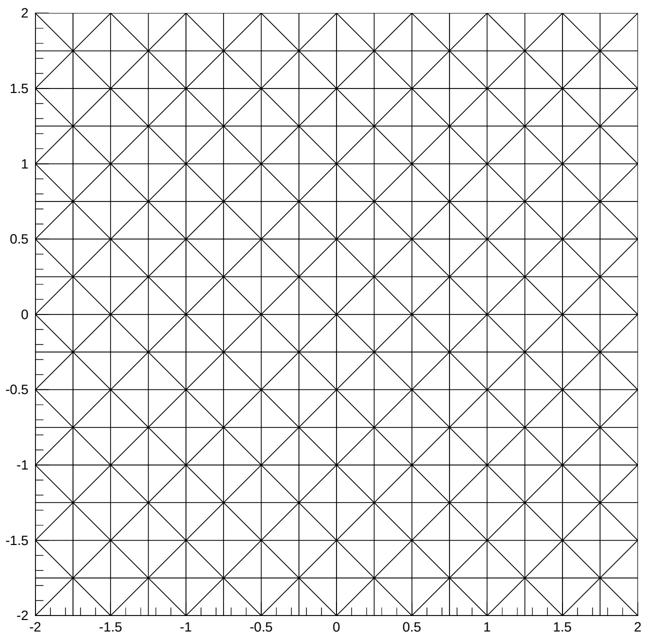

FIG. 4.6. *Triangulation for two dimensional Burgers equation,* $h = \frac{1}{4}$.

Example 4.8. The problem of a propagating surface:

$$
\left\{
\begin{array}{ll}
\varphi_t - (1 - \varepsilon K)\sqrt{1 + \varphi_x^2 + \varphi_y^2} = 0, & 0 < x < 1,\ 0 < y < 1 \\
\varphi(x, y, 0) = 1 - \frac{1}{4}(\cos(2\pi x) - 1)(\cos(2\pi y) - 1) &
\end{array}
\right.
\tag{4.8}
$$

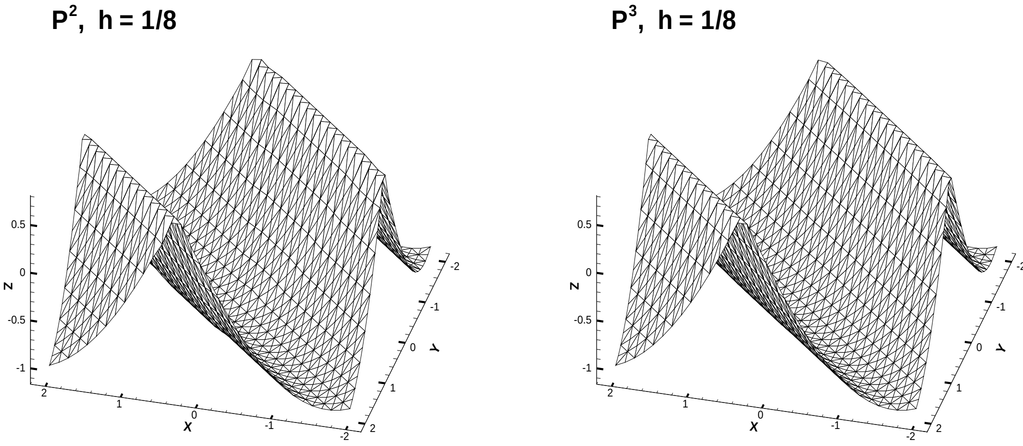

FIG. 4.7. *Two dimension Burgers' equation, triangular mesh, t=1.5/π^2.*

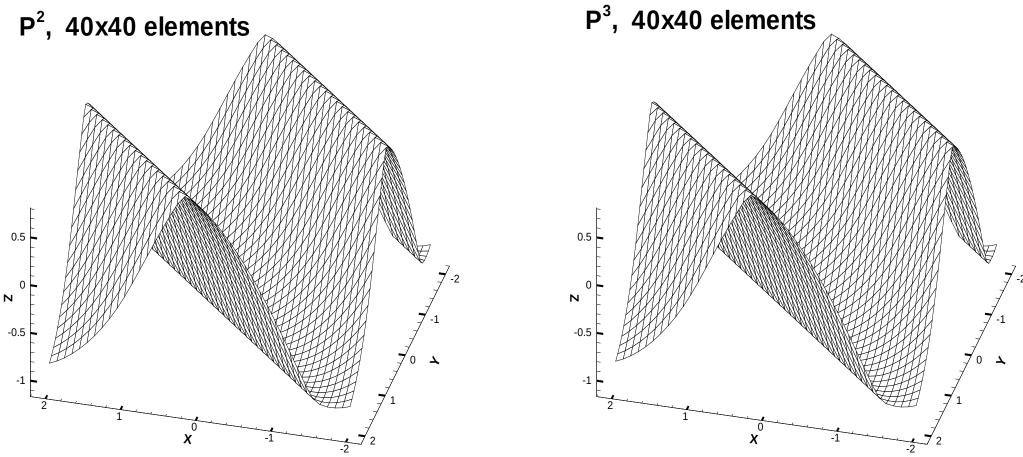

FIG. 4.8. *Two dimensional,* $H(u,v) = -\cos(u+v+1), t = 1.5/\pi^2$.

where K is the mean curvature defined by

$$K = -\frac{\varphi_{xx}(1+\varphi_y^2) - 2\varphi_{xy}\varphi_x\varphi_y + \varphi_{yy}(1+\varphi_x^2)}{(1+\varphi_x^2+\varphi_y^2)^{\frac{3}{2}}}, \tag{4.9}$$

and ε is a small constant. Periodic boundary condition is used.

This problem was studied in [23] by using the finite difference ENO schemes. We apply the discontinuous Galerkin method, with the second derivative terms handled by the local discontinuous Galerkin techniques presented and analyzed in [11], which

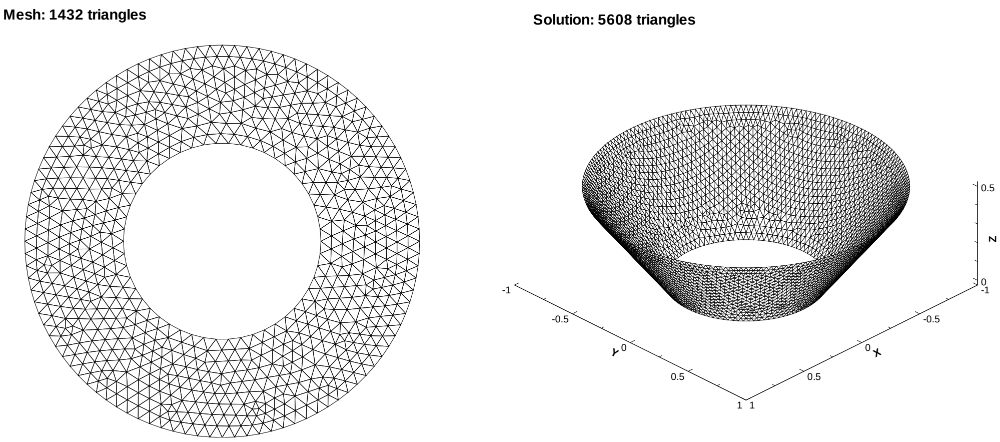

FIG. 4.9. *The level set equation,* P^2.

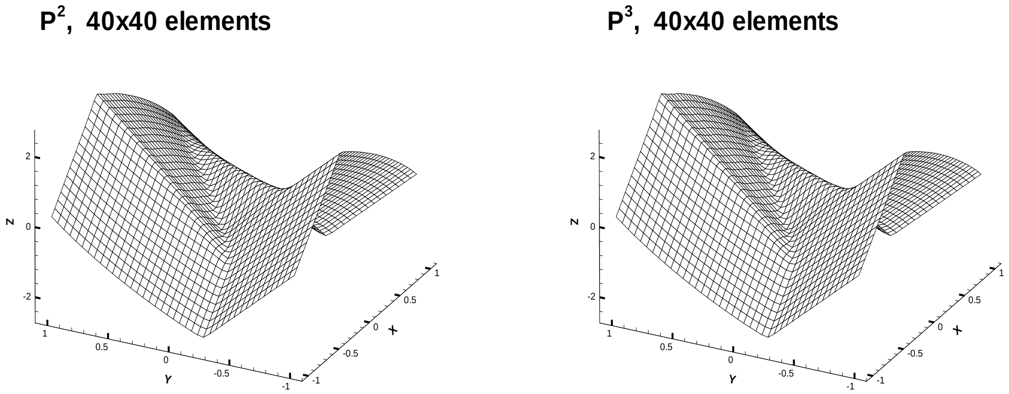

FIG. 4.10. *Two dimensional Riemann problem,* $H(u,v) = \sin(u+v)$, $t = 1$.

amounts to solving the following system

$$\left\{\begin{array}{l} u_t - \left(\sqrt{1+u^2+v^2} + \varepsilon\frac{p(1+v^2)-2quv+r(1+u^2)}{1+u^2+v^2}\right)_x = 0 \\ v_t - \left(\sqrt{1+u^2+v^2} + \varepsilon\frac{p(1+v^2)-2quv+r(1+u^2)}{1+u^2+v^2}\right)_y = 0 \\ p - u_x = 0 \\ q - u_y = 0 \\ r - v_y = 0 \end{array}\right. \tag{4.10}$$

using the discontinuous Galerkin method. The details of the method, especially the choices of fluxes, which are important for stability, can be found in [11].

We first use a uniform rectangular mesh of 50×50 elements and the local Lax-Friedrichs flux (3.8)-(3.9). The results of $\varepsilon = 0$ (pure convection) and $\varepsilon = 0.1$ are presented in Fig. 4.11 and Fig. 4.12, respectively. Notice that the surface at $t = 0$ is shifted downward by 0.35 in order to show the detail of the solution at $t = 0.3$.

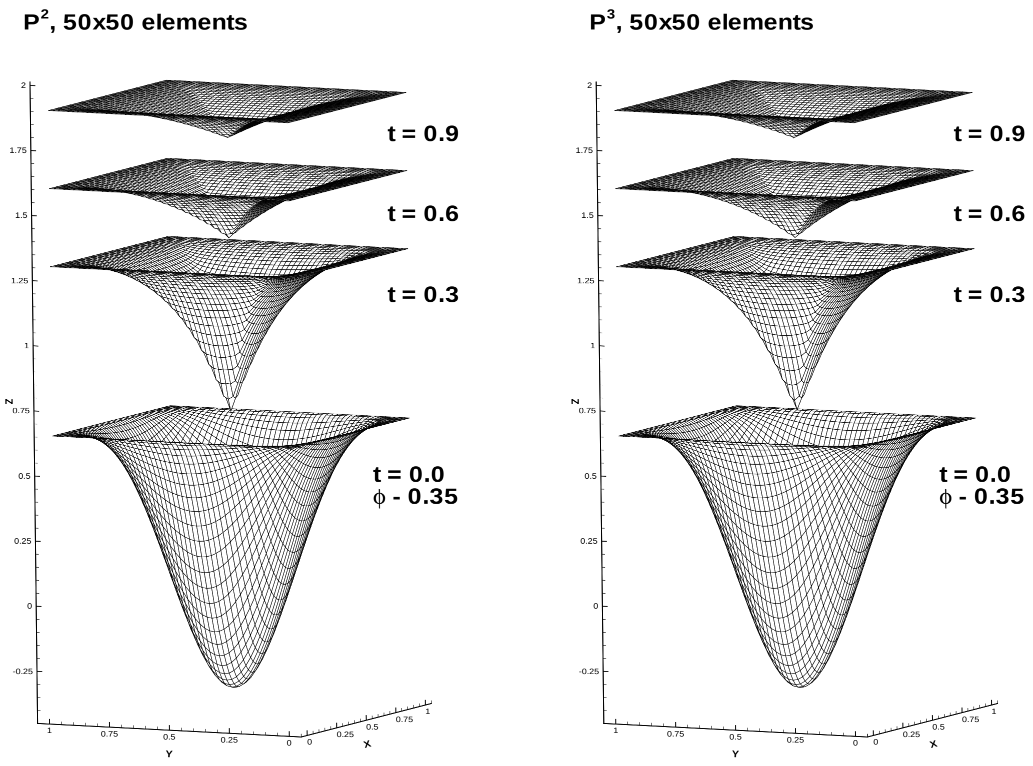

FIG. 4.11. *Propagating surfaces, rectangular mesh,* $\varepsilon = 0$.

Next we use a triangulation shown in Fig. 4.13. We refine the mesh around the center of domain where the solution develops discontinuous derivatives (for the $\varepsilon = 0$ case). There are 2146 triangles and 1108 nodes in this triangulation. The solutions are displayed in Fig. 4.14 and Fig. 4.15, respectively, for $\varepsilon = 0$ (pure convection) and $\varepsilon = 0.1$. Notice that we again shift the solution at $t = 0.0$ downward by 0.35 to show the detail of the solutions at later time.

Example 4.9. The problem of a propagating surface on a unit disk. The equation is the same as (4.8) in the previous example, but it is solved on a unit disk $x^2 + y^2 < 1$ with an initial condition

$$\varphi(x, y, 0) = \sin\left(\frac{\pi(x^2 + y^2)}{2}\right)$$

and a Neumann type boundary condition $\nabla\varphi = 0$.

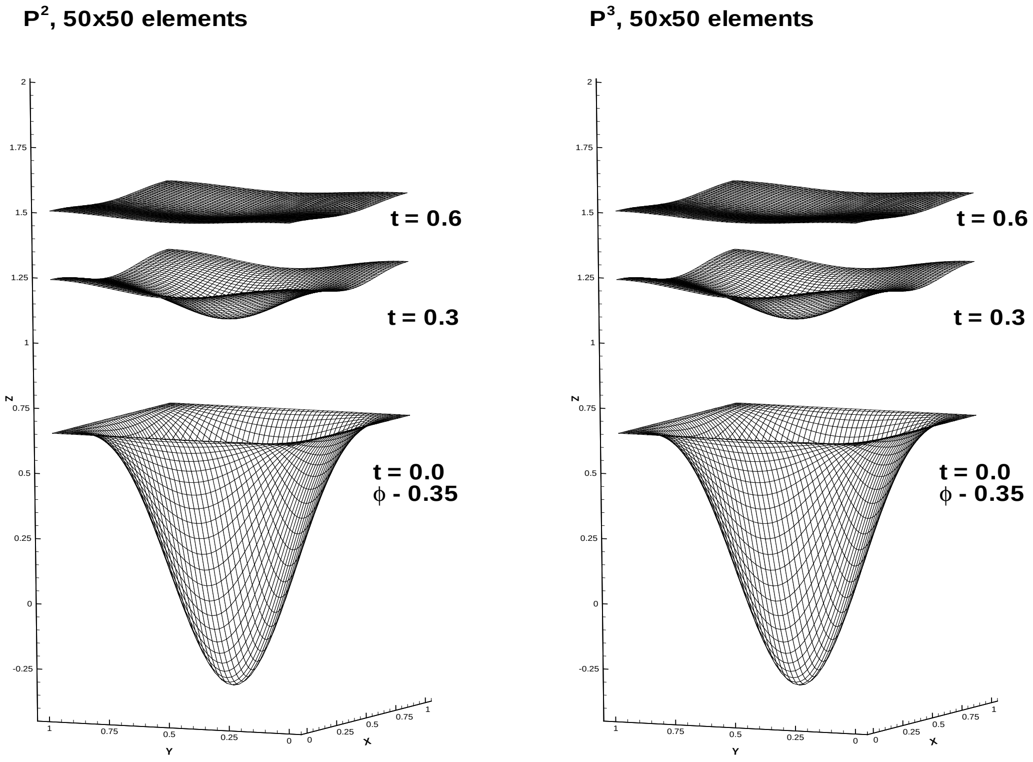

FIG. 4.12. *Propagating surfaces, rectangular mesh,* $\varepsilon = 0.1$.

It is difficult to use rectangular meshes for this problem. Instead we use the triangulation shown in Fig. 4.16. Notice that we have again refined the mesh near the center of the domain where the solution develops discontinuous derivatives. There are 1792 triangles and 922 nodes in this triangulation. The solutions with $\varepsilon = 0$ are displayed in Fig. 4.17. Notice that the solution at $t = 0$ is shifted downward by 0.2 to show the detail of the solution at later time.

The solution with $\varepsilon = 0.1$ are displayed in Fig. 4.18. Notice that the solution at $t = 0$ is again shifted downward by 0.2 to show the detail of the solution at later time.

Example 4.10. A problem from optimal control [24]:

$$\left\{ \begin{array}{l} \varphi_t + (\sin y)\varphi_x + (\sin x + \mathrm{sign}(\varphi_y))\varphi_y - \frac{1}{2}\sin^2 y - (1-\cos x) = 0 \\ \varphi(x, y, 0) = 0 \end{array} \right. \tag{4.11}$$

in the domain $-\pi < x < \pi$, $-\pi < y < \pi$ with periodic boundary conditions. We use a uniform rectangular mesh of 40×40 elements and the local Lax-Friedrichs flux (3.8)-(3.9). The solution at $t = 1$ is shown in Fig. 4.19, while the optimal control $w = \mathrm{sign}(\varphi_y)$ is shown in Fig. 4.20.

Notice that our method computes $\nabla\varphi$ as an independent variable. It is very desirable for those problems in which the most interesting features are contained in the first derivatives of φ, as in this optimal control problem.

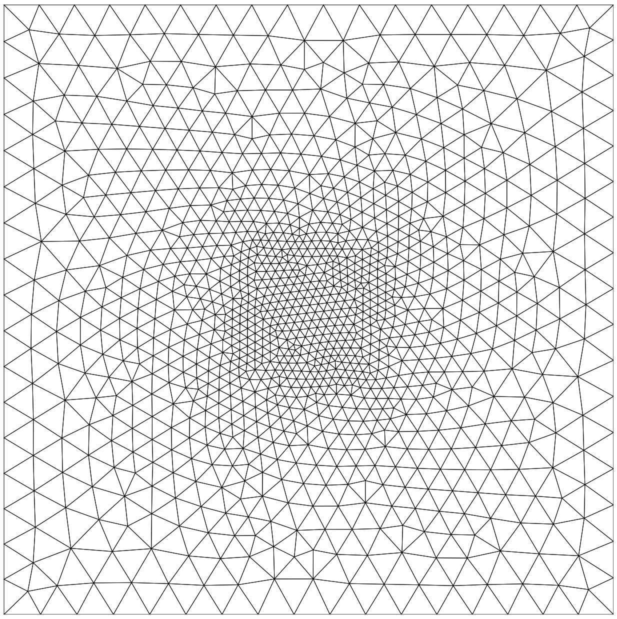

FIG. 4.13. *Triangulation used for the propagating surfaces.*

Example 4.11. A problem from computer vision [25]:

$$\left\{ \begin{array}{ll} \varphi_t + I(x,y)\sqrt{1+\varphi_x^2+\varphi_y^2} - 1 = 0, & -1 < x < 1, -1 < y < 1 \\ \varphi(x,y,0) = 0 & \end{array} \right. \tag{4.12}$$

with $\varphi = 0$ as the boundary condition. The steady state solution of this problem is the shape lighted by a source located at infinity with vertical direction. The solution is not unique if there are points at which $I(x,y) = 1$. Conditions must be prescribed at those points where $I(x,y) = 1$. Since our method is a finite element method, we need to prescribe suitable conditions at the correspondent elements. We take

$$I(x,y) = 1/\sqrt{1+(1-|x|)^2+(1-|y|)^2} \tag{4.13}$$

The exact steady solution is $\varphi(x,y,\infty) = (1-|x|)(1-|y|)$. We use a uniform rectangular mesh of 40×40 elements and the local Lax-Friedrichs flux (3.8)-(3.9). We impose the exact boundary conditions for $u = \varphi_x, v = \varphi_y$ from the above exact steady solution, and take the exact value at one point (the lower left corner) to recover φ. The results for P^2 and P^3 are presented in Fig. 4.21, while Fig. 4.22 contains the history of iterations to the steady state.

Next we take

$$I(x,y) = 1/\sqrt{1+4y^2(1-x^2)^2+4x^2(1-y^2)^2} \tag{4.14}$$

The exact steady solution is $\varphi(x,y,\infty) = (1-x^2)(1-y^2)$. We again use a uniform rectangular mesh of 40×40 elements, the local Lax-Friedrichs flux (3.8)-(3.9), impose the exact boundary conditions for $u = \varphi_x, v = \varphi_y$ from the above exact steady solution, and take the exact value at one point (the lower left corner) to recover φ. A

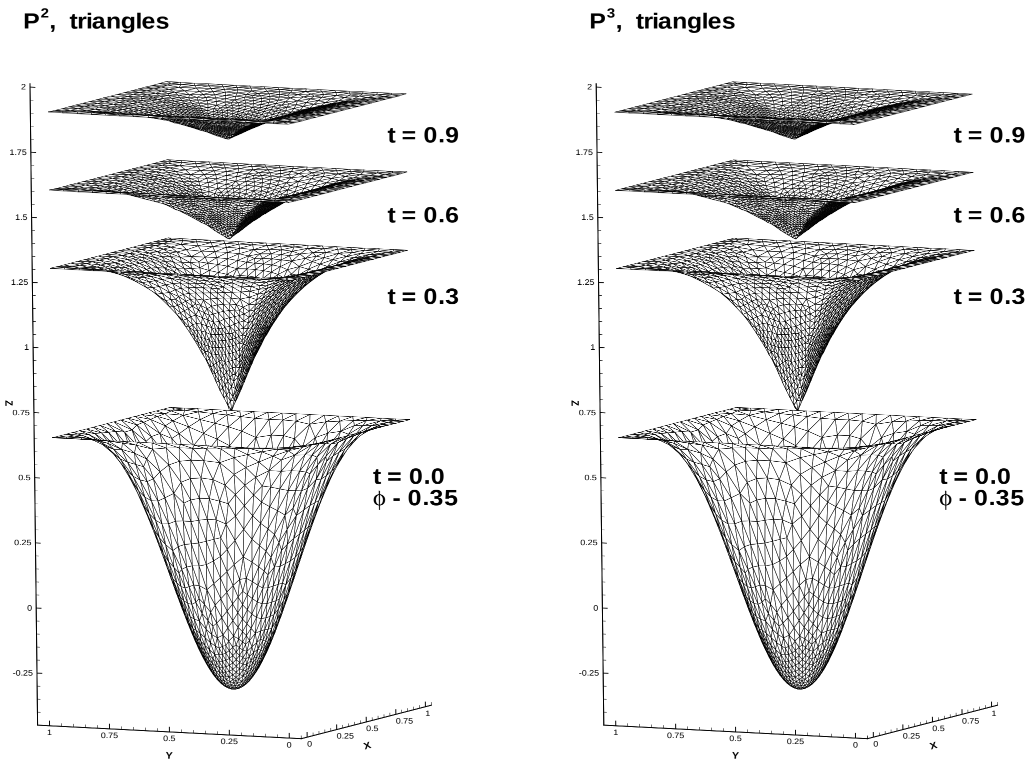

FIG. 4.14. *Propagating surfaces, triangular mesh, $\varepsilon = 0$.*

continuation method is used, with the steady solution using

$$I_\varepsilon(x, y) = 1/\sqrt{1 + 4y^2(1 - x^2)^2 + 4x^2(1 - y^2)^2 + \varepsilon} \tag{4.15}$$

for bigger ε as the initial condition for smaller ε. The sequence of ε used are $\varepsilon = 0.2, 0.05, 0$. The results for P^2 and P^3 are presented in Fig. 4.23.

5. Concluding Remarks.

We have developed and tested a class of discontinuous Galerkin methods for solving nonlinear Hamilton-Jacobi equations. These methods have the advantage of easy handling of complicated geometry and local mesh refinements, as well as efficient parallel implementations. Numerical examples in one and two space dimensions are shown to illustrate the capability of the methods.

REFERENCES

[1] R. Abgrall, *On essentially non-oscillatory schemes on unstructured meshes: analysis and implementation*, J. Comput. Phys., v114 (1994), pp.45–58.

[2] R. Abgrall, *Numerical discretization of the first-order Hamilton-Jacobi equation on triangular meshes.*, Comm. Pure Appl. Math., v49 (1996), pp.1339–1373.

[3] T. Barth and J. Sethian, *Numerical schemes for the Hamilton-Jacobi and level set equations on triangulated domains*, J. Comput. Phys., v145 (1998), pp.1–40.

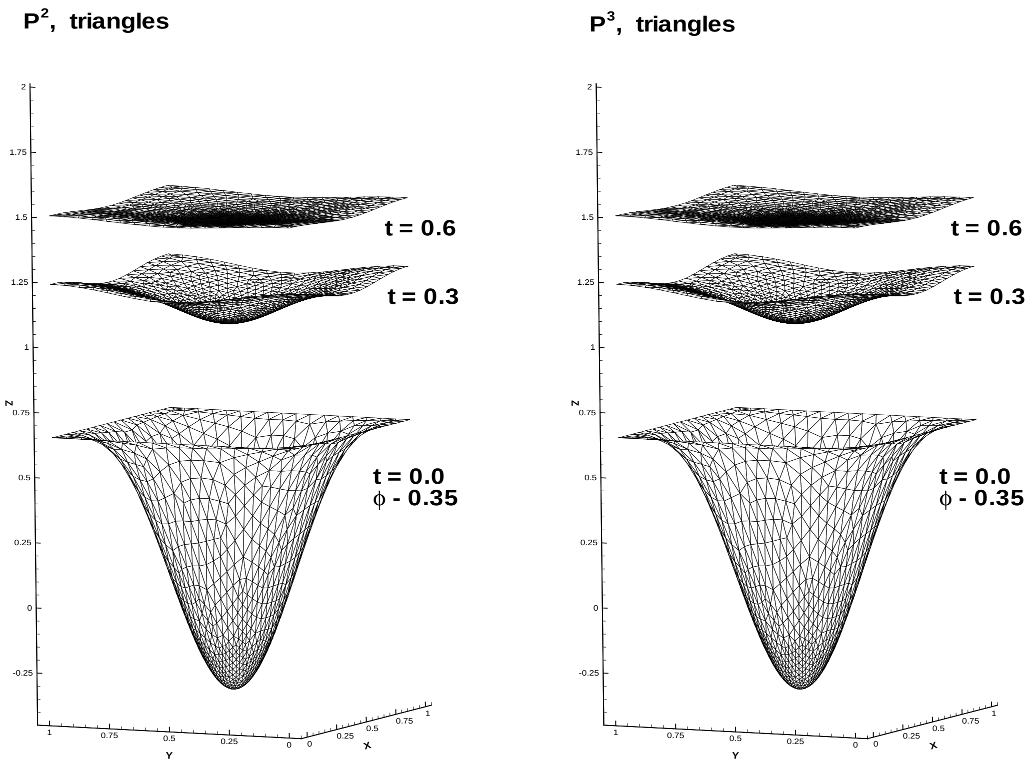

FIG. 4.15. *Propagating surfaces, triangular mesh,* $\varepsilon = 0.1$.

[4] R. Biswas, K.D. Devine and J. Flaherty, *Parallel, adaptive finite element methods for conservation laws*, Appl. Numer. Math., v14 (1994), pp.255–283.

[5] B. Cockburn, *An introduction to the discontinuous Galerkin method for convection-dominated problems*, in *Advanced Numerical Approximation of Nonlinear Hyperbolic Equations*, A. Quarteroni, Editor, Lecture Notes in Mathematics, CIME subseries, Springer-Verlag, to appear.

[6] B. Cockburn and C.-W. Shu, *The Runge-Kutta local projection P^1-discontinuous Galerkin method for scalar conservation laws*, M^2AN, v25 (1991), pp.337–361.

[7] B. Cockburn and C.-W. Shu, *TVB Runge-Kutta local projection discontinuous Galerkin finite element method for scalar conservation laws II: general framework*, Math. Comp., v52 (1989), pp.411–435.

[8] B. Cockburn, S.-Y. Lin and C.-W. Shu, *TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws III: one dimensional systems*, J. Comput. Phys., v84 (1989), pp.90–113.

[9] B. Cockburn, S. Hou, and C.-W. Shu, *TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws IV: the multidimensional case*, Math. Comp., v54 (1990), pp.545–581.

[10] B. Cockburn and C.-W. Shu, *TVB Runge-Kutta local projection discontinuous Galerkin finite element method for scalar conservation laws V: multidimensional systems*, J. Comput. Phys., v141 (1998), pp.199-224.

[11] B. Cockburn and C.-W. Shu, *The local discontinuous Galerkin method for time-dependent convection diffusion systems*, SIAM J. Numer. Anal., to appear.

[12] M. Crandall and P.L. Lions, *Viscosity solutions of Hamilton-Jacobi equations*, Trans. Amer. Math. Soc., v277 (1983), pp.1–42.

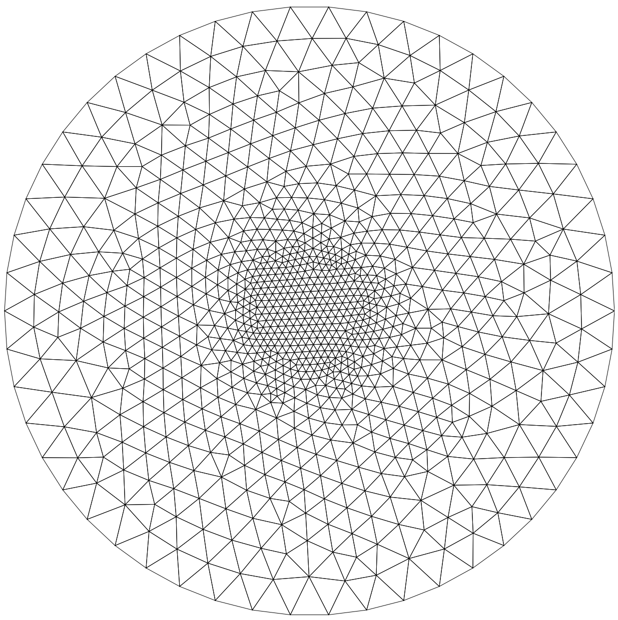

FIG. 4.16. *Triangulation for the propagating surfaces on a disk.*

[13] M. Crandall and P.L. Lions, *Two approximations of solutions of Hamilton-Jacobi equations*, Math. Comp., v43 (1984), pp.1–19.

[14] S. Gottlieb and C.-W. Shu, *Total variation diminishing Runge-Kutta schemes*, Math. Comp., v67 (1998), pp.73-85.

[15] A. Harten, B. Engquist, S. Osher and S. Chakravarthy, *Uniformly high order essentially non-oscillatory schemes, III*, J. Comput. Phys., v71 (1987), pp.231–303.

[16] G. Jiang and D. Peng, *Weighted ENO schemes for Hamilton-Jacobi equations*, SIAM J. Sci. Comput., to appear.

[17] G. Jiang and C.-W. Shu, *On cell entropy inequality for discontinuous Galerkin methods*, Math. Comp., v62 (1994), pp.531–538.

[18] G. Jiang and C.-W. Shu, *Efficient implementation of weighted ENO schemes*, J. Comput. Phys., v126 (1996), pp.202–228.

[19] S. Jin and Z. Xin *Numerical passage from systems of conservation laws to Hamilton-Jacobi equations*, SIAM J. Numer. Anal., to appear.

[20] F. Lafon and S. Osher, *High order two dimensional nonoscillatory methods for solving Hamilton-Jacobi scalar equations*, J. Comput. Phys., v123 (1996), pp.235–253.

[21] X.-D. Liu, S. Osher and T. Chan, *Weighted essentially nonoscillatory schemes*, J. Comput. Phys., v115 (1994), pp.200–212.

[22] I. Lomtev and G. Karniadakis, *A discontinuous spectral/hp element Galerkin method for the Navier-Stokes equations*, Int. J. Num. Meth. Fluids, to appear.

[23] S. Osher and J. Sethian, *Fronts propagating with curvature dependent speed: algorithms based on Hamilton-Jacobi formulations*, J. Comput. Phys., v79 (1988), pp.12–49.

[24] S. Osher and C.-W. Shu, *High-order essentially nonoscillatory schemes for Hamilton-Jacobi equations*, SIAM J. Numer. Anal., v28 (1991), pp.907–922.

[25] E. Rouy and A. Tourin, *A viscosity solutions approach to shape-from- shading*, SIAM J. Numer. Anal., v29 (1992), pp.867–884.

[26] C.-W. Shu and S. Osher, *Efficient implementation of essentially non-oscillatory shock capturing schemes*, J. Comput. Phys., v77 (1988), pp.439–471.

[27] C.-W. Shu and S. Osher, *Efficient implementation of essentially non-oscillatory shock capturing schemes II*, J. Comput. Phys., v83 (1989), pp.32–78.

[28] M. Sussman,P. Smereka, and S. Osher, *A level set approach for computing solution to incompressible two-phase flow*, J. Comput. Phys., v114 (1994), pp.146–159.

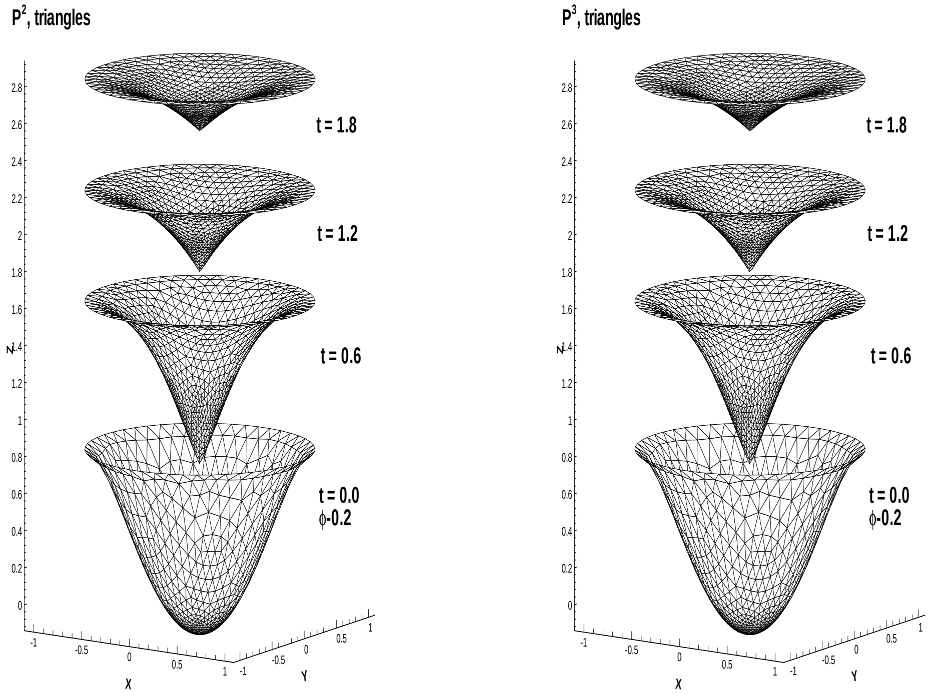

FIG. 4.17. *Propagating surfaces on a disk, triangular mesh,* $\varepsilon = 0$.

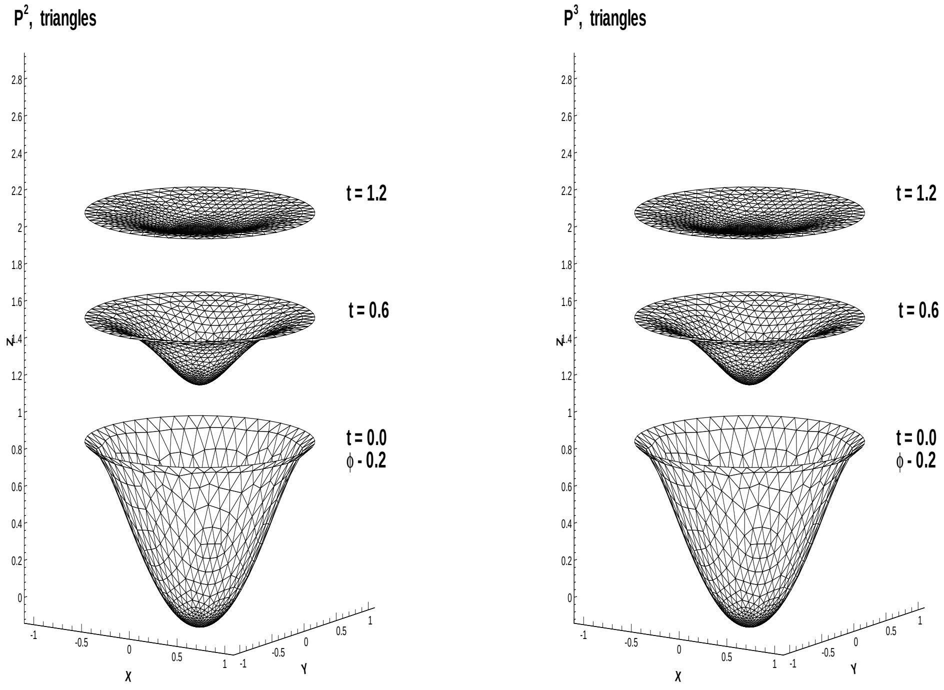

FIG. 4.18. *Propagating surfaces on a disk, triangular mesh,* $\varepsilon = 0.1$.

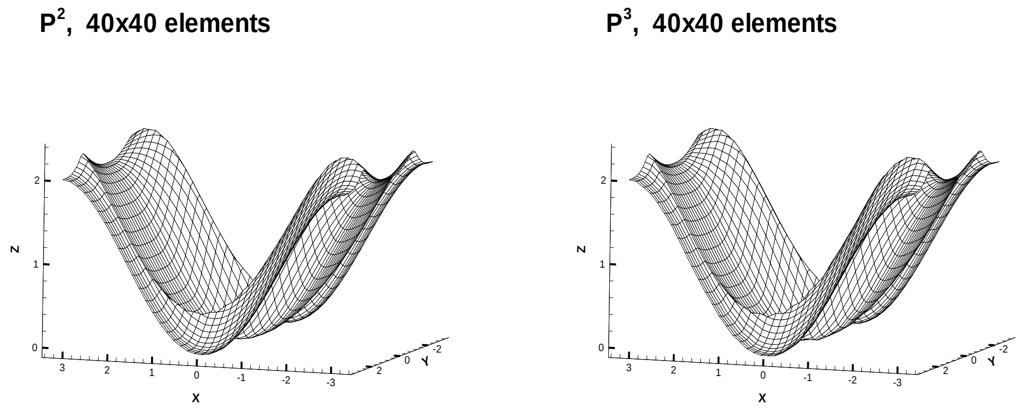

FIG. 4.19. *Control problem,* $t = 1$.

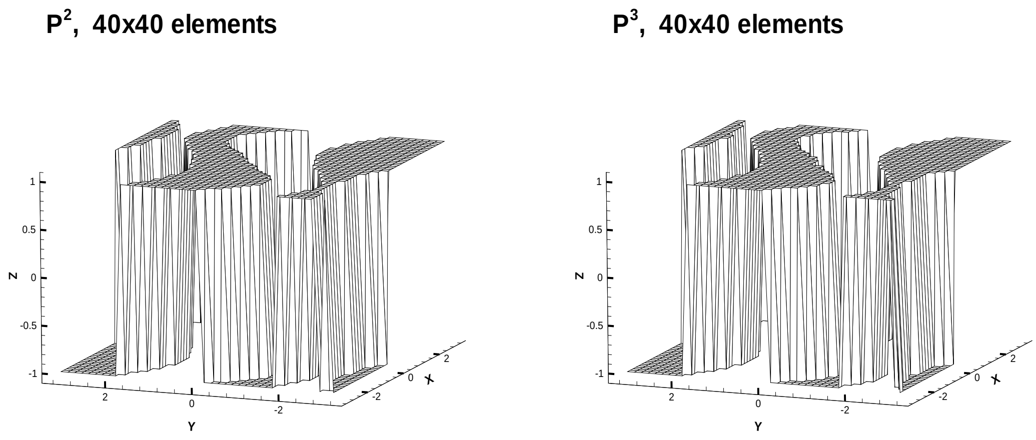

FIG. 4.20. *Control problem,* $t = 1, w = \mathrm{sign}(\varphi_y)$.

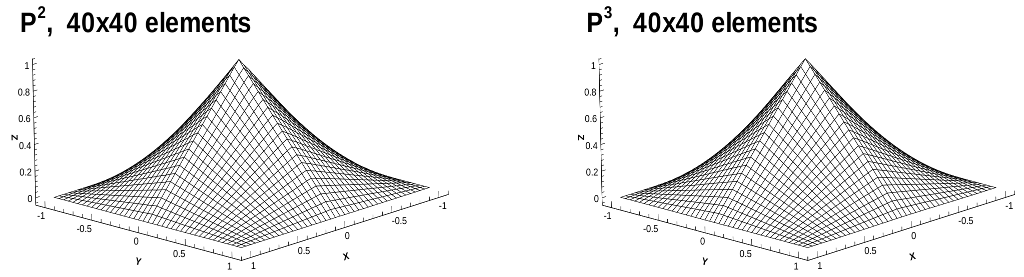

FIG. 4.21. *Computer vision problem,* $\varphi(x, y, \infty) = (1 - |x|)(1 - |y|)$.

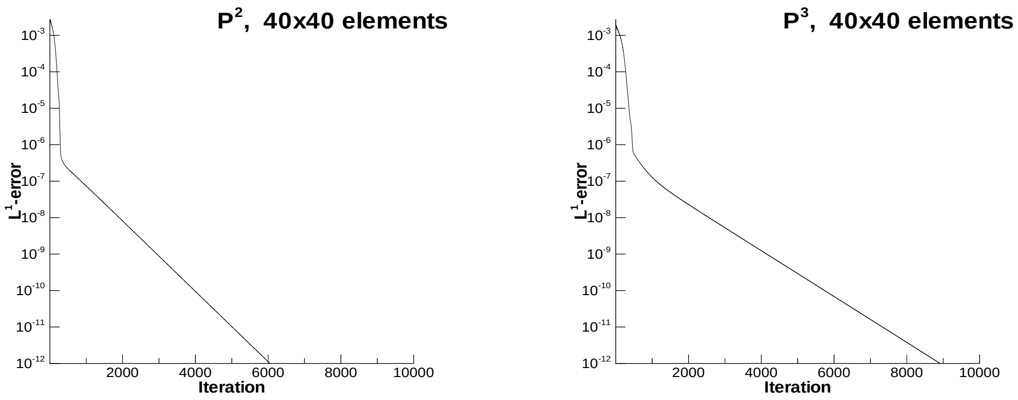

FIG. 4.22. *Computer vision problem, history of iterations.*

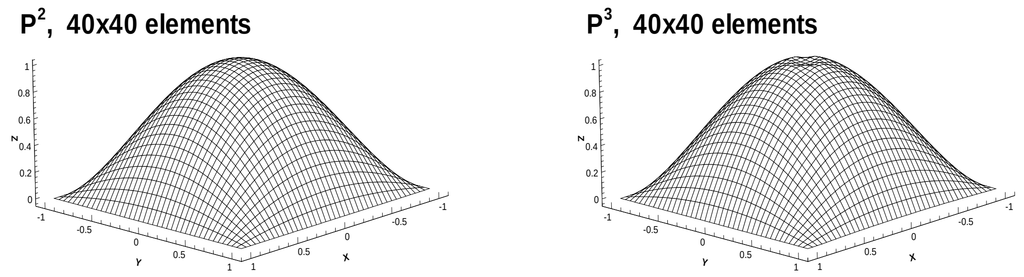

FIG. 4.23. *Computer vision problem,* $\varphi(x, y, \infty) = (1 - x^2)(1 - y^2)$.