SIAM J. NUMER. ANAL.
Vol. 47, No. 1, pp. 675–698

THE DIRECT DISCONTINUOUS GALERKIN (DDG) METHODS FOR DIFFUSION PROBLEMS*

HAILIANG LIU† AND JUE YAN†

Abstract. A new discontinuous Galerkin finite element method for solving diffusion problems is introduced. Unlike the traditional local discontinuous Galerkin method, the scheme called the direct discontinuous Galerkin (DDG) method is based on the direct weak formulation for solutions of parabolic equations in each computational cell and lets cells communicate via the numerical flux $\widehat{u_x}$ *only*. We propose a general numerical flux formula for the solution derivative, which is consistent and conservative; and we then introduce a concept of admissibility to identify a class of numerical fluxes so that the nonlinear stability for both one-dimensional (1D) and multidimensional problems are ensured. Furthermore, when applying the DDG scheme with admissible numerical flux to the 1D linear case, kth order accuracy in an energy norm is proven when using kth degree polynomials. The DDG method has the advantage of easier formulation and implementation and efficient computation of the solution. A series of numerical examples are presented to demonstrate the high order accuracy of the method. In particular, we study the numerical performance of the scheme with different admissible numerical fluxes.

Key words. diffusion, discontinuous Galerkin methods, stability, convergence rate, numerical trace

AMS subject classifications. 65M12, 65M60

DOI. 10.1137/080720255

1. Introduction.

In this paper, we introduce a new discontinuous Galerkin (DG) method for solving nonlinear diffusion equations of the form

$$\partial_t U - \nabla \cdot (A(U)\nabla U) = 0, \quad \Omega \times (0,T), \tag{1.1}$$

where $\Omega \subset \mathbb{R}^d$, the matrix $A(U) = (a_{ij}(U))$ is symmetric and positive definite, and U is an unknown function of (x,t).

The novelty of our method is to use the direct weak formulation for solutions of (1.1) in each computational cell and let cells communicate through a numerical trace of $A(U)\nabla U$ only. It is from this feature that the method proposed here derives its name: the direct DG (DDG) method. Here we carefully design a class of numerical fluxes in such a way that a stable and high order accurate DG method for the nonlinear diffusion equation (1.1) is achieved.

The DG method is a finite element method using a completely discontinuous piecewise polynomial space for the numerical solution and the test functions. A key ingredient of this method is the suitable design of the interelement boundary treatments (the so-called numerical fluxes) to obtain high order accurate and stable schemes. The DG method has been vigorously developed for hyperbolic problems since it was first introduced in 1973 by Reed and Hill [25] for neutron transport equations. A major development of the DG method is carried out by Cockburn, Shu, and collaborators in a series of papers [16, 15, 14, 11, 18] for nonlinear hyperbolic

*Received by the editors April 4, 2008; accepted for publication (in revised form) September 9, 2008; published electronically January 16, 2009.
http://www.siam.org/journals/sinum/47-1/72025.html
†Mathematics Department, Iowa State University, Ames, IA 50011 (hliu@iastate.edu, jyan@iastate.edu). The first author's research was supported by the National Science Foundation under grant DMS05-05975.

conservation laws. While it is being actively developed, the DG method has found rapid applications in many areas; we refer to [20, 13, 19] for further references.

However, the DG method when applied to diffusion problems encounters subtle difficulties, which can be illustrated by the simple one-dimensional (1D) heat equation

$$u_t - u_{xx} = 0.$$

Indeed, using this equation in [28], Shu illustrated some typical "pitfalls" in using the DG method for viscous terms. The DG method when applied to the heat equation formally leads to

$$\int_{I_j} u_t v + \int_{I_j} u_x v_x dx - \widehat{(u_x)}_{j+1/2} v^-_{j+1/2} + \widehat{(u_x)}_{j-1/2} v^+_{j-1/2} = 0, \tag{1.2}$$

where both u and v are piecewise polynomials on each computational cell $I_j = (x_{j-1/2}, x_{j+1/2})$. Notice that u itself is discontinuous at cell interfaces; the formulation (1.2) even requires approximations of u_x at cell interfaces, which we call the numerical flux $\widehat{(u_x)}$!

A primary choice is the slope average $\widehat{(u_x)}_{j+1/2} = ((u_x)^-_{j+1/2} + (u_x)^+_{j+1/2})/2$. But the scheme produces a completely incorrect, therefore inconsistent, solution; see Figure 1 (left). This is called "subtle inconsistency" by Shu in [28].

There are two ways to remedy this problem which were suggested in the literature. One is to rewrite the heat equation into a 1st order system and solve it with the DG method

$$u_t - q_x = 0, \quad q - u_x = 0.$$

Here both u and the auxiliary variable q are evolved in each computational cell. This method was originally proposed for the compressible Navier–Stokes equation by Bassi and Rebay [4]. Subsequently, a generalization called the local discontinuous Galerkin (LDG) method was introduced in [17] by Cockburn and Shu and further studied in [10, 7, 12, 8]. More recently, the LDG methods have been successfully extended to higher order partial differential equations; see, e.g., [32, 22, 31, 23].

Another one is to add extra cell boundary terms so that a weak stability property is ensured. The scheme thus takes the following form:

$$\begin{aligned}&\int_{I_j} u_t v + \int_{I_j} u_x v_x dx - \widehat{(u_x)}_{j+1/2} v^-_{j+1/2} + \widehat{(u_x)}_{j+1/2} v^+_{j-1/2} \\ &- \frac{1}{2}(v_x)^-_{j+1/2}\left(u^+_{j+1/2} - u^-_{j+1/2}\right) - \frac{1}{2}(v_x)^+_{j-1/2}\left(u^+_{j-1/2} - u^-_{j-1/2}\right) = 0,\end{aligned}$$

where again the slope average was chosen as the numerical flux. Such a method was introduced by Baumann and Oden [5]; see also Oden, Babuska, and Baumann [24]. This later scheme, once written into a primal formulation, is similar to a class of *interior penalty* methods, independently proposed and studied for elliptic and parabolic problems in the 1970s; see, e.g., [2, 3, 30]. Considering the similarities among the recently introduced DG methods, Arnold et al. [1] have set the existing DG methods into a unified framework with a systematic analysis of these methods via linear elliptic problems. Another framework using both the equation in each element and continuity relations across interfaces was recently analyzed in [6].

Notice that the above two ways suggest modifications mainly on the scheme formulation but not on the numerical flux $\widehat{u_x}$. The main goal of this work is to propose

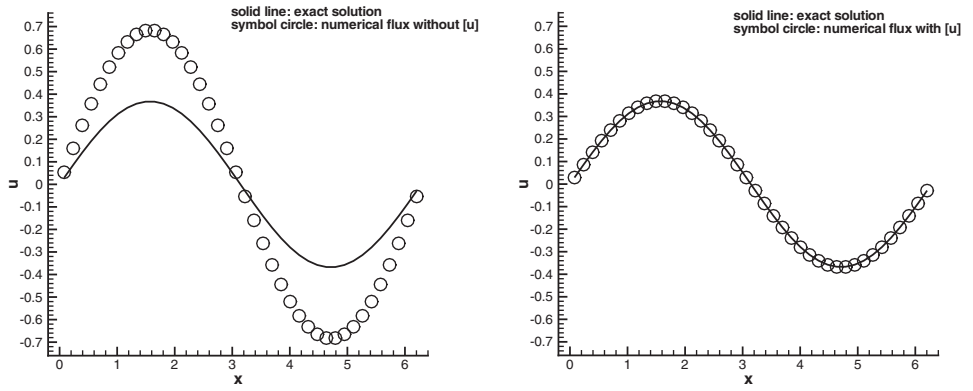

FIG. 1. *On the left* $\widehat{u_x} = \overline{u_x}$ *and on the right* $\widehat{u_x} = \frac{[u]}{\Delta x} + \overline{u_x}$ *at* $t = 1$*, mesh size* $N = 40$*.* p^1 *polynomial approximation.*

a path which sticks to the direct weak formulation (1.2) but with new choices of numerical flux $\widehat{u_x}$ to obtain a stable and accurate DG scheme. More precisely, the heart of the DDG method is to use the direct weak formulation for parabolic equations and let cells communicate via the numerical flux $\widehat{u_x}$. A key observation is that the jump of the function itself relative to the mesh size, when numerically measuring slopes of a discontinuous function, plays an essential role. For example, for the piecewise constant approximation ($k = 0$), the choice of

$$\widehat{u_x} = \frac{u^+ - u^-}{\Delta x}$$

leads to the standard central finite difference scheme. When we use the numerical flux

$$\widehat{u_x} = \frac{u^+ - u^-}{\Delta x} + \frac{1}{2}\left(u_x^+ + u_x^-\right),$$

the resulting scheme with piecewise linear approximation is found of 2nd order accurate and of course gives the correct solution; see Figure 1 (right).

However, the trace of the solution derivative under a diffusion process is rather subtle. From the PDE point of view, jumps of all even order derivatives as well as the average of odd order derivatives all contribute to the trace of the solution derivative. We propose a general numerical flux formula, which is consistent with the solution gradient and conservative. The form of the numerical flux is motivated by an exact trace formulation derived from solving the heat equation with smooth initial data having only one discontinuous point.

We then introduce a concept of admissibility for numerical fluxes. The admissibility condition serves as a criterion for selecting suitable numerical fluxes to guarantee nonlinear stability of the DDG method and corresponding error estimates. Indeed in the linear case, the convergence rate of order $(\Delta x)^k$ for the error in a parabolic energy norm $L^\infty(0,T;L^2) \cap L^2(0,T;H^1)$ is obtained when p^k polynomials are used.

In this paper, we restrict ourselves to diffusion problems with periodic boundary conditions. We shall display the most distinctive features of the DDG method using as simple a setting as possible. This paper is organized as follows. In section 2, we introduce the DDG methods for the 1D problems. For this model problem, the main

idea of how to devise the method is presented. The nonlinear stability and error estimate for the linear case are discussed in section 3. In section 4, we extend the DDG methods to multidimensional problems in which U is a scalar and $A = (a_{ij})_{d\times d}$ is a positive and semidefinite matrix. The nonlinear stability is established. Finally in section 5, we present a series of numerical results to validate our DDG methods. For completeness some projection properties and a trace formula for the heat equation are presented in the appendix.

Finally, we note that formulating a DG method without rewriting the equation into a 1st order system as in the LDG method was also explored in three more recent works [29, 21, 9]. But they all rely on repeated integration by parts for the diffusion term so that the interface values can be imposed for both the solution and its derivatives. In contrast, we use the standard weak formulation for parabolic equations with integration by parts only once, and the interface continuity is enforced by defining suitable interface values of the solution derivative only.

2. 1D diffusion process. In this section, we introduce the formulation of the DDG method for the simple 1D case

$$U_t - (a(U)U_x)_x = 0 \quad \text{in } (0,1)\times(0,T) \tag{2.1}$$

subject to initial data

$$U(x,0) = U_0(x) \quad \text{on } (0,1) \tag{2.2}$$

and periodic boundary conditions.

The unknown function U is a scalar, and we assume the diffusion coefficient a to be a nonnegative function of U. The DDG method is constructed upon the direct weak formulation of parabolic equations.

First, we partition the domain $(0,1)$ by grid points $0 = x_{1/2} < x_{3/2} < \cdots < x_{N+1/2} = 1$; we define the mesh $\{I_j = (x_{j-1/2}, x_{j+1/2}), \quad j = 1\cdots N\}$ and set the mesh size $\Delta x_j = x_{j+1/2} - x_{j-1/2}$. Furthermore, we denote $\Delta x = \max_{1\le j\le N}\Delta x_j$. We seek an approximation u to U such that for any time $t\in[0,T]$, $u\in\mathbb{V}_{\Delta x}$,

$$\mathbb{V}_{\Delta x} := \left\{ v\in L^2(0,1): \quad v|_{I_j}\in P^k(I_j), \quad j = 1,\dots,N \right\},$$

where $P^k(I_j)$ denotes the space of polynomials in I_j with degree at most k. We now formulate our scheme for (2.1) and describe guidelines for defining numerical fluxes.

2.1. Formulation of the scheme. Denote the flux $h := h(U,U_x) = a(U)U_x$. Let U be the exact solution of the underlying problem. Multiply (2.1) by any smooth function $V\in H^1(0,1)$, integrate on I_j, and have integration by parts to obtain the following equations:

$$\int_{I_j} U_t V dx - h_{j+1/2}V_{j+1/2} + h_{j-1/2}V_{j-1/2} + \int_{I_j} a(U)U_xV_x dx = 0, \tag{2.3}$$

$$\int_{I_j} U(x,0)V(x)dx = \int_{I_j} U_0(x)V(x)dx. \tag{2.4}$$

Here the time derivative is to be understood in the weak sense, and $h_{j\pm1/2}$ and $V_{j\pm1/2}$ denote values of h and V at $x = x_{j\pm1/2}$, respectively.

Next we replace the smooth function V by any test function $v\in\mathbb{V}_{\Delta x}$ and the exact solution U by the numerical approximate solution u. The flux $h(U,U_x)$ is replaced by the numerical flux $\widehat{h}$ that will be defined later.

Thus the approximate solution given by the DDG method is defined as

$$\int_{I_j} u_t v dx - \widehat{h}_{j+1/2} v^-_{j+1/2} + \widehat{h}_{j-1/2} v^+_{j-1/2} + \int_{I_j} a(u) u_x v_x dx = 0, \tag{2.5}$$

$$\int_{I_j} u(x,0) v(x) dx = \int_{I_j} U_0(x) v(x) dx. \tag{2.6}$$

Note that u is a well defined function since there are as many equations per element as unknowns. The integral $\int_{I_j} a(u) u_x v_x dx$ could be either computed exactly or approximated by using suitable numerical quadratures. Thus, to complete the DG space discretization, we only have to define the numerical flux $\widehat{h}$.

2.2. The numerical flux. Crucial for the stability as well as for the accuracy of the DDG method is the choice of the numerical flux $\widehat{h}$. To define it, we adopt the following notations:

$$u^{\pm}(t) = u\left(x^{\pm}_{j+1/2}, t\right), \quad [u] = u^+ - u^-, \quad \overline{u} = \frac{u^+ + u^-}{2}.$$

The numerical flux $\widehat{h}$ defined at the cell interface $x_{j+1/2}$ is chosen in such a way that it is a function depending only on the left and right polynomials and that it (i) is consistent with $h = b(u)_x = a(u)u_x$, where $b(u) = \int^u a(s)ds$ when u is smooth; (ii) is conservative in the sense of $\widehat{h}$ being single valued on $x_{j+1/2}$ and

$$\frac{d}{dt}\int_{I_j} u dx = \widehat{h}_{j+1/2} - \widehat{h}_{j-1/2};$$

(iii) ensures the L^2-stability; and (iv) enforces the high order accuracy of the method.

Motivated by the trace formula of the solution derivative of the heat equation, see (7.3) in the appendix, we propose the following general format of the numerical flux:

$$\widehat{h} = D_x b(u) = \beta_0 \frac{[b(u)]}{\Delta x} + \overline{b(u)_x} + \sum_{m=1}^{\lfloor k/2 \rfloor} \beta_m (\Delta x)^{2m-1} \left[\partial_x^{2m} b(u)\right], \tag{2.7}$$

where k is the highest degree of polynomials in two adjacent computational cells and $\lfloor \cdot \rfloor$ is the floor function. Note here in (2.7) and in what follows that for nonuniform mesh Δx should be replaced by $(\Delta x_j + \Delta x_{j+1})/2$ and for uniform mesh $\Delta x = 1/N$.

The numerical flux $\widehat{h}$, which is an approximation of $b(U)_x$ at the cell interface, involves the average $\overline{b(u)_x}$ and the jumps of even order derivatives of $b(u)$, $[\partial_x^{2m} b(u)]$, up to $m = \lfloor k/2 \rfloor$. For example, with the p^3 polynomial approximation we need to determine suitable β_0 and β_1 to define the numerical flux

$$\widehat{h} = D_x b(u) = \beta_0 \frac{[b(u)]}{\Delta x} + \overline{b(u)_x} + \beta_1 \Delta x [b(u)_{xx}].$$

It is clear for any choice of β_i's that the numerical flux defined in (2.7) is consistent and conservative. As is known, the underlying solution for the heat equation is smooth, and thus jumps of discrete solutions across cell interfaces have to be properly controlled so that continuities can be enforced at least in a weak sense.

To ensure stability and enhance accuracy and more importantly to measure the goodness of the choice of β_i's, we introduce a notion of admissibility for numerical fluxes as follows.

DEFINITION 2.1 (admissibility). *We call a numerical flux* $\widehat{h}$ *of the form* (2.7) admissible *if there exists a* $\gamma \in (0,1)$ *and* $\alpha > 0$ *such that*

$$\gamma \sum_{j=1}^{N} \int_{I_j} a(u) u_x^2(x,t)dx + \sum_{j=1}^{N} \widehat{h}_{j+1/2}[u]_{j+1/2} \geq \alpha \sum_{j=1}^{N} \frac{([b(u)][u])_{j+1/2}}{\Delta x} \tag{2.8}$$

holds for any piecewise polynomials of degree k, *i.e.,* $u \in \mathbb{V}_{\Delta x}$.

It is shown in the next section that for any admissible flux the DDG scheme is nonlinear stable and has kth order accuracy in an energy norm when using p^k polynomials for linear problems. We note that for error analysis $\alpha > 0$ plays an essential role in controlling the total jumps across cell interfaces.

We now discuss some principles for finding β_i's. To simplify the presentation we restrict our discussions to the linear case with $\widehat{h} = D_x u$.

For the piecewise constant approximation, $k = 0$, the numerical flux (2.7) reduces to

$$\widehat{u_x} = D_x u = \beta_0 \frac{[u]}{\Delta x}.$$

Clearly we should take $\beta_0 = 1$, for which the DDG scheme is consistent with the central finite difference scheme. Note that $\beta_0 \neq 1$ is admissible but gives $O(1)$ error.

For the piecewise linear approximation, $k = 1$, the numerical flux (2.7) with $\beta_0 = 1$ becomes

$$D_x u = \frac{[u]}{\Delta x} + \overline{u_x}. \tag{2.9}$$

This can be easily verified to be admissible with $\alpha = 1/2$ and $\gamma = 1/2$. The corresponding DDG scheme is of $2nd$ order as observed numerically in section 5.

We can now prove that (2.9), with possibly an additional amount of $[u]/\Delta x$, is admissible for polynomial approximations of any degree, even for nonlinear diffusion.

THEOREM 2.1. *Consider the* 1*D diffusion with* $a(u) \geq \delta > 0$. *The numerical flux*

$$D_x u = \beta_0 \frac{[b(u)]}{\Delta x} + \overline{u}_x \tag{2.10}$$

is admissible for any piecewise polynomial of degree $k \geq 0$ *provided* β_0 *is suitably large.*

Proof. It is sufficient to select β_0 so that the underlying flux is admissible locally around each cell, i.e.,

$$\gamma \int_{I_j} a(u) u_x^2 dx + D_x u[u] \geq \alpha [b(u)][u]/\Delta x,$$

which, when combined with (2.10), can be rewritten as

$$\gamma \Delta x \int_{I_j} a(u) u_x^2 dx + \overline{u}_x [u] \Delta x + (\beta_0 - \alpha) \frac{[b(u)]}{[u]} [u]^2 \geq 0.$$

Note for $k = 0$, $\beta(0) = 1$ is admissible for $\alpha \le 1$. From $a(u) \ge \delta$ we have $\frac{[b(u)]}{[u]} \ge \delta$. Thus the above inequality is ensured to hold for all $u|_{I_j} \in P^k(I_j)$ and therefore for all $[u]$, provided

$$(\overline{u}_x \Delta x)^2 - 4(\beta_0 - \alpha)\gamma \Delta x \delta \int_{I_j} u_x^2 dx \le 0.$$

Summing this inequality over all indexes $j \in N$ we have

$$\beta_0 \ge \alpha + \frac{1}{4\gamma\delta} \frac{\Delta x \sum_j \overline{u}_x^2}{\sum_j \int_{I_j} a(u) u_x^2 dx}.$$

Maximizing the right side over all $u|_{I_j} \in P^k(I_j)$ we obtain

$$\beta_0 \ge \alpha + \frac{M_k}{4\gamma\delta^2},$$

where

$$M_k = \max_{u \in P^k(I_j)} \frac{\Delta x \sum_j \overline{u}_x^2}{\sum_j \int_{I_j} u_x^2 dx}.$$

For example, when $a(u) = 1$, $M_0 = 0$, $M_1 = 1$, $M_2 = 3$, etc. ∎

Numerical experiments show that the scheme with numerical flux (2.10) achieves $(k+1)$th order accuracy if k is odd but kth order accuracy if k is even, as long as β_0 is chosen above a critical value $\beta^* \sim M_k$ (to guarantee the scheme stability). The scheme accuracy is not sensitive to the choice of β_0, though the critical value β^* needs to be larger as k increases.

In order to gain the $(k+1)$th order accuracy when k is even it is necessary to use higher order derivatives within our DDG framework. We consider exploring higher order approximations. The idea is to construct a higher order polynomial $\tilde{p}(x) \in P^{k+1}(I_j \cup I_{j+1})$ across the interface by interpolating at sample points in two neighboring cells. There are $[k/2] + 1$ pairs of points symmetrically sampled on each side of the underlying interface. Then the numerical flux can be defined as

$$D_x u = \partial_x \tilde{p}(x)|_{x_{j+1/2}}. \tag{2.11}$$

For $k = 2, 3$ we explore the Stirling interpolation formula based on four symmetric points

$$x_{j+1/2} \pm \frac{1}{2}h, \quad x_{j+1/2} \pm h, \quad 0 < h \le \Delta x,$$

leading to a unique 3rd order polynomial, whose derivative when evaluated at the cell interface $x_{j+1/2}$ gives

$$D_x u = \frac{7}{6}\frac{[u]}{h} + \overline{u_x} + \frac{h}{12}[u_{xx}]. \tag{2.12}$$

For p^2 and p^3 polynomials, the numerical flux (2.12) with $h = \Delta x$ enables us to obtain the optimal 3rd and 4th orders of accuracy, respectively. This suggests that the step used in the Stirling interpolation spans exactly the full computational cell on each

side, no more and no less; it is also unbiased. Of course, for nonuniform mesh, Δx needs to be understood as $(\Delta x_j + \Delta x_{j+1})/2$.

Here we note yet another way to select β_1 for $k \le 2$ based on the exact trace formula (7.3), i.e.,

$$u_x(0,t) = \frac{1}{\sqrt{4\pi t}}[u] + \overline{u_x} + \sqrt{\frac{t}{\pi}}[u_{xx}], \quad 0 < t < \Delta t.$$

Considering the parabolic scaling, the correct mesh ratio should be $t \sim (\Delta x)^2$. Therefore setting $t = (\eta \Delta x)^2$ we obtain the following numerical flux:

$$D_x u = \frac{1}{\sqrt{4\pi}} \frac{[u]}{\eta \Delta x} + \overline{u_x} + \frac{\eta \Delta x}{\sqrt{\pi}}[u_{xx}].$$

In section 5, we carry out numerical experiments for these η-schemes, and the choice $\eta = \sqrt{\pi}/12$, i.e., again $\beta_1 = 1/12$, gives the best performance, both in the absolute error and the order of the scheme.

In summary for $p^k, k = 0, \ldots, 3$, we advocate the DDG scheme with the following numerical flux:

$$D_x u = \frac{[u]}{\Delta x} + \overline{u}_x + \frac{\Delta x}{12}[u_{xx}]. \tag{2.13}$$

For p^k with $k \ge 4$ we employ the simple flux (2.10). It is interesting to note that for the p^2 case the coefficient $\beta_1 = 1/12$ is indeed important, but the β_0 is less important in the sense that with other choices of β_0 3rd order accuracy can also be achieved. In comparison, for the p_0 case $\beta_0 = 1$ is important.

Remark 2.1. The recipe given in (2.11) leads to a class of admissible numerical fluxes (2.12). But numerically only the flux with $h = \Delta x$ delivers the optimal L^2 accuracy for P^2 element, which is also the case for both nonuniform meshes in the 1D setting and hypercube partitions in multidimensions; for the latter see (5.10) in Example 5.5. This fact is further illustrated in Example 5.4 when the equation is nonlinear.

2.3. Time discretization. Up to now, we have taken the method of lines approach and have left t continuous. For time discretization we can use total variation diminishing (TVD) high order Runge–Kutta methods [27, 26] to solve the method of lines ODE

$$u_t = L(u). \tag{2.14}$$

The 3rd order TVD Runge–Kutta method that we use in this paper is given by

$$\begin{aligned} u^{(1)} &= u^n + \Delta t L(u^n), \\ u^{(2)} &= \frac{3}{4}u^n + \frac{1}{4}u^{(1)} + \frac{1}{4}\Delta t L\left(u^{(1)}\right), \\ u^{n+1} &= \frac{1}{3}u^n + \frac{2}{3}u^{(2)} + \frac{2}{3}\Delta t L\left(u^{(2)}\right). \end{aligned}$$

3. The nonlinear stability and error estimates.

3.1. The nonlinear stability. We first review the stability property for the continuous problem. We let $U \in L^2$ be a smooth solution to the initial value problem (2.1)–(2.2). Set $V = U$ in the weak formulation, and integrate over $[0, T]$; we have the following energy identity:

$$\frac{1}{2}\int_0^1 U^2(x,T)dx + \int_0^T \int_0^1 a(U)U_x^2 dx d\tau = \frac{1}{2}\int_0^1 U_0^2 dx \quad \forall\, T > 0.$$

We say the DDG scheme is L^2 stable if the numerical solution $u(x,t)$ satisfies

$$\int_0^1 u^2(x,T)dx \le \int_0^1 U_0^2 dx.$$

In fact, the numerical solution defined by our DDG scheme (2.5) and (2.6) not only satisfies this stability property but also has total control on all jumps crossing cell interfaces $\{x_{j+1/2}\}_{j=1}^N$ due to the admissibility of the numerical flux.

THEOREM 3.1 (energy stability). *Consider the DDG scheme* (2.5) and (2.6) *with numerical flux* (2.7). *If the numerical flux is admissible as described in* (2.8), *then*

$$\begin{aligned}\frac{1}{2}\int_0^1 u^2(x,T)dx + (1-\gamma)\int_0^T \sum_{j=1}^N \int_{I_j} a(u)u_x^2(x,t)dxdt \\ + \alpha \int_0^T \sum_{j=1}^N \frac{[b(u)]}{\Delta x}[u]dt \le \frac{1}{2}\int_0^1 U_0^2(x)dx.\end{aligned} \tag{3.1}$$

Proof. Setting $v = u$ in (2.5), we have

$$\frac{d}{dt}\int_{I_j} \frac{u^2}{2}dx + \int_{I_j} a(u)u_x^2 dx - \widehat{h}_{j+1/2}u^-_{j+1/2} + \widehat{h}_{j-1/2}u^+_{j-1/2} = 0.$$

Summation over $j = 1, 2 \cdots N$ and integration with respect to t over $[0, T]$ leads to

$$\begin{aligned}\frac{1}{2}\int_0^1 u^2(x,T)dx + \int_0^T \sum_{j=1}^N \int_{I_j} a(u)u_x^2(x,t)dxdt \\ + \int_0^T \sum_{j=1}^N \widehat{h}_{j+1/2}[u]_{j+1/2}dt = \frac{1}{2}\int_0^1 u^2(x,0)dx.\end{aligned} \tag{3.2}$$

From the admissible condition (2.8) of the numerical flux $\widehat{h}_{j+1/2}$ defined in (2.7) it follows that

$$\begin{aligned}\int_0^T \sum_{j=1}^N \widehat{h}_{j+1/2}[u]_{j+1/2}dt = \int_0^T \sum_{j=1}^N \left(\widehat{b(u)_x}[u]\right)_{j+1/2} dt \\ \ge \alpha \int_0^T \sum_{j=1}^N \frac{[b(u)]}{\Delta x}[u]dt - \gamma \int_0^T \sum_{j=1}^N \int_{I_j} a(u)u_x^2(x,t)dxdt.\end{aligned} \tag{3.3}$$

Finally, we note that (2.6) with $v(x) = u(x,0)$ gives

$$\int_0^1 u^2(x,0)dx \le \int_0^1 U_0^2(x)dx. \tag{3.4}$$

Insertion of (3.3) and (3.4) into (3.2) leads to the desired stability estimate (3.1). ∎

Notice that the usual L^2-stability follows from such a stability estimate (3.1) since the term $[b(u)][u] = a(u^*)[u]^2$ remains nonnegative for any jumps $[u]$.

3.2. Error estimates. Now we turn to the question of the quality of the approximate solution defined by the DDG method. In the linear case $a(u) = 1$, from the above stability result and from the approximation properties of the finite element space $\mathbb{V}_{\Delta x}$, we can estimate the error $e := u - U$ between the exact solution U and the numerical solution u. Inspired by the stability estimate (3.1) we introduce the following energy norm to measure the solutions and the error:

$$|||v(\cdot,t)||| := \left(\int_0^1 v^2 dx + (1-\gamma) \int_0^t \sum_{j=1}^N \int_{I_j} v_x^2 dx d\tau + \alpha \int_0^t \sum_{j=1}^N \frac{[v]^2}{\Delta x} d\tau \right)^{1/2}, \tag{3.5}$$

with $\gamma \in (0,1)$ and $\alpha > 0$. From the stability analysis and the smoothness of the exact solution U, we reformulate stability estimates for both the exact solution and the numerical solution in terms of the norm $|||(\cdot,T)|||$:

$$|||U(\cdot,T)||| \le |||U(\cdot,0)|||, \quad |||u(\cdot,T)||| \le |||U(\cdot,0)|||.$$

This section is devoted to the proof of the following error estimate.

THEOREM 3.2 (error estimate). *Let U be the exact solution and e be the error between the exact solution and the numerical solution by the DDG method with numerical flux* (2.7). *If the numerical flux is admissible* (2.8), *then the energy norm of the error satisfies the inequality*

$$|||e(\cdot,T)||| \le C|||\partial_x^{k+1} U(\cdot,T)|||(\Delta x)^k, \tag{3.6}$$

where $C = C(k,\gamma,\alpha)$ is a constant depending on k, γ, and α but is independent of U and Δx.

Remark 3.1. The error estimates are optimal in Δx for smooth solutions. For initial data in $H^{k+1}(0,1)$ we can simply replace $|||\partial_x^{k+1} U(\cdot,T)|||$ by $|U_0|_{k+1}$ since for parabolic problem $U_t = U_{xx}$ we have

$$\frac{1}{2}\int_0^1 \left|\partial_x^{k+1} U(x,T)\right|^2 dx + \int_0^T \int_0^1 \left|\partial_x^{k+2} U(x,T)\right|^2 dx \le \frac{1}{2}\int_0^1 \left|\partial_x^{k+1} U_0(x)\right|^2 dx, \tag{3.7}$$

which holds for solution U with initial data $U_0 \in H^{k+1}(0,1)$.

Remark 3.2. The kth order energy error (3.6) does not automatically imply a $(k+1)$th order L^2 error estimate unless the scheme is adjoint-consistent; see, e.g., [1]. The inclusion of jumps of higher order derivatives in numerical flux in this paper is intended to restore the optimal L^2 error.

Let $\mathbb{P}$ be the L^2 projection operator from $H^1(0,1)$ to the finite element space $\mathbb{V}_{\Delta x}$, which is defined as the only polynomial $\mathbb{P}(U)(x)$ in $\mathbb{V}_{\Delta x}$ such that

$$\int_{I_j} (\mathbb{P}(U)(x) - U(x))v(x)dx = 0 \qquad \forall v \in \mathbb{V}_{\Delta x}.$$

Note by (2.6) and the above L^2 projection definition we have that $u(x,0) = \mathbb{P}(U_0)$.

To estimate $e = u - U$, we rewrite the error as

$$e = u - \mathbb{P}(U) + \mathbb{P}(U) - U = \mathbb{P}(e) - (U - \mathbb{P}(U)). \tag{3.8}$$

Thus we have

$$|||e(\cdot,T)||| \le |||\mathbb{P}(e)(\cdot,T)||| + |||(U-\mathbb{P}(U))(\cdot,T)|||. \tag{3.9}$$

It suffices to estimate the two terms on the right. The projection properties are essentially used and summarized in the following auxiliary lemma. The proof of the lemma is based on Bramble–Hilbert lemma 7, and an extended discussion is postponed in the appendix.

LEMMA 3.1 (L^2 projection properties). *Let $U \in H^{s+1}(I_j)$ for $j = 1, \dots, N$ and $s \ge 0$. Then we have the following estimates:*

1. $|\mathbb{P}(U) - U|_{m,I_j} \le c_k(\Delta x)^{(\min\{k,s\}+1-m)}|U|_{s+1,I_j}, \quad m \le k+1.$
2. $|\partial_x^m(\mathbb{P}(U) - U)_{x_{j+1/2}}| \le c_k(\Delta x)^{(\min\{k,s\}+1/2-m)}|U|_{s+1,I_{j+1/2}}, \quad m \le k+1/2,$

where $m \ge 0$ is an integer, $I_{j+1/2} = I_j \bigcup I_{j+1}$, and constant c_k depends on k but is independent of I_j and U; $|\cdot|_{m,I_j}$ denotes the seminorm of $H^m(I_j)$.

These basic estimates enable us to prove the following lemma.

LEMMA 3.2. *Let U be the smooth exact solution, and for any function v the $D_x v$ at the cell interface $x_{j+1/2}$ is defined by*

$$D_x v = \overline{v_x} + \sum_{m=0}^{\lfloor k/2 \rfloor} \beta_m (\Delta x)^{2m-1}[\partial_x^{2m} v]. \tag{3.10}$$

Then we have

(i) *projection error*

$$|||(U - \mathbb{P}(U))(\cdot,T)||| \le C|||\partial_x^{k+1}U|||(\Delta x)^k,$$

(ii) *trace error*

$$\int_0^T \sum_{j=1}^N (D_x(U - \mathbb{P}(U)))^2_{j+1/2}(t)dt \le C|||\partial_x^{k+1}U|||^2(\Delta x)^{2k-1}.$$

Proof. (i) Apply the estimates in Lemma 3.1 to $|||(U - \mathbb{P}(U))(\cdot,T)|||^2$ to obtain

$$\sum_{j=1}^N |\mathbb{P}(U) - U|^2_{0,I_j} + (1-\gamma)\int_0^T \sum_{j=1}^N |(\mathbb{P}(U) - U)|^2_{1,I_j}dt + \alpha \int_0^T \sum_{j=1}^N \frac{[\mathbb{P}(U) - U]^2}{\Delta x}dt$$
$$\le C_k \left((\Delta x)^{2k+2}|U|^2_{k+1,[0,1]} + (\Delta x)^{2k}\int_0^T |U(\cdot,t)|^2_{k+2,[0,1]}dt \right).$$

Thus the estimate in (i) is ensured.

(ii) Applying the estimates in Lemma 3.1 to the expression (3.10) with $v = U - \mathbb{P}(U)$ we have

$$\begin{aligned}
&\sum_{j=1}^N (D_x(U - \mathbb{P}(U)))^2_{j+1/2} \\
&\le C_k \sum_{j=1}^N \left\{ (\Delta x)^{2k-1}|U|^2_{k+2,I_j} + \sum_{m=0}^{\lfloor k/2 \rfloor} (\Delta x)^{4m-2}(\Delta x)^{2k+1-4m}|U|^2_{k+2,I_{j+1/2}} \right\} \\
&\le C(\Delta x)^{2k-1}|U|^2_{k+2,[0,1]}.
\end{aligned}$$

This gives the estimate (ii). The proof is thus complete. ∎

To finish the estimate of e in (3.9), it remains to estimate $\mathbb{P}(e)$, which favorably lies in $\mathbb{V}_{\Delta x}$.

LEMMA 3.3. *We have*

$$|||\mathbb{P}(e)(\cdot,T)|||^2 \le \frac{1}{(1-\gamma)^2}|||(U-\mathbb{P}(U))(\cdot,T)|||^2 + \frac{\Delta x}{\alpha}\int_0^T \sum_{j=1}^N (D_x(U-\mathbb{P}(U)))^2_{j+1/2}dt.$$

A combination of Lemmas 3.2 and 3.3 with the inequality (3.9) yields the desired estimate (3.6), which completes the proof of Theorem 4.1.

We now conclude this section by presenting a detailed proof of Lemma 3.3.

Proof of Lemma 3.3. First, we define a bilinear form $\mathbb{B}(w,v)$ as

(3.11)
$$\mathbb{B}(w,v) = \int_0^T\int_0^1 w_t(x,t)v(x,t)dx + \int_0^T\sum_{j=1}^N\int_{I_j} w_x(x,t)v_x(x,t)dxdt + \Theta(T,\widehat{w_x},v)$$

for any $v \in \mathbb{V}_{\Delta x}$, and

$$\Theta(T,\widehat{w_x},v) = \int_0^T \sum_{j=1}^N \left(\widehat{(w_x)}_{j+1/2}[v]_{j+1/2}\right)dt. \tag{3.12}$$

By the definition of DDG scheme (2.5), we have $\mathbb{B}(u,v)=0\ \forall v\in\mathbb{V}_{\Delta x}$. Exact solution $U(x,t)$ also satisfies $\mathbb{B}(U,v)=0\ \forall v\in\mathbb{V}_{\Delta x}$, and then we have

$$\mathbb{B}(e,v) = \mathbb{B}(u-U,v) = 0.$$

This equality when combined with (3.8) gives

$$\mathbb{B}(\mathbb{P}(e),v) = \mathbb{B}(U-\mathbb{P}(U),v).$$

Taking $v = u-\mathbb{P}(U) = \mathbb{P}(e)$, we have

$$\mathbb{B}(\mathbb{P}(e),\mathbb{P}(e)) = \mathbb{B}(U-\mathbb{P}(U),\mathbb{P}(e)). \tag{3.13}$$

Note the left-hand side of the equality involves the term $\mathbb{P}(e)$ that we want to estimate. The right-hand side of the equality is $\mathbb{B}(U-\mathbb{P}(U),\mathbb{P}(e))$, which is expected to be small because it involves the error between the exact solution and its L^2 projection $U-\mathbb{P}(U)$.

Letting $w=v=\mathbb{P}(e)$ in (3.11) and using $\mathbb{P}(e)(\cdot,0)=0$, we have

(3.14)
$$\mathbb{B}(\mathbb{P}(e),\mathbb{P}(e)) = \frac{1}{2}\|\mathbb{P}(e)(\cdot,T)\|^2 + \int_0^T\sum_{j=1}^N \|(\mathbb{P}(e))_x(\cdot,t)\|^2_{I_j}dt + \Theta\left(T,\widehat{(\mathbb{P}(e))_x},\mathbb{P}(e)\right).$$

Recalling the definition of admissibility for the numerical flux in (2.7) and the interface contribution term Θ defined in (3.12), we obtain

$$\Theta\left(T,\widehat{(\mathbb{P}(e))_x},\mathbb{P}(e)\right) \ge \alpha\int_0^T\sum_{j=1}^N\frac{[\mathbb{P}(e)]^2}{\Delta x}dt - \gamma\int_0^T\sum_{j=1}^N\|(\mathbb{P}(e))_x(\cdot,t)\|^2_{I_j}dt.$$

Hence

$$\mathbb{B}(\mathbb{P}(e),\mathbb{P}(e)) \ge |||\mathbb{P}(e)(\cdot,T)|||^2 - \frac{1}{2}||\mathbb{P}(e)(\cdot,T)||^2. \tag{3.15}$$

On the other hand,

$$\mathbb{B}(U-\mathbb{P}(U),\mathbb{P}(e)) = \int_0^T \int_0^1 (U-\mathbb{P}(U))_t \mathbb{P}(e)dxdt$$

$$+\int_0^T \sum_{j=1}^N \int_{I_j} (U-\mathbb{P}(U))_x (\mathbb{P}(e))_x dxdt + \Theta\left(T, \widehat{(U-\mathbb{P}(U))}_x, \mathbb{P}(e)\right). \tag{3.16}$$

With $\mathbb{P}(e) \in \mathbb{V}_{\Delta x}$, we have

$$\int_0^T \int_0^1 (U-\mathbb{P}(U))_t \mathbb{P}(e)dxdt = 0.$$

For the second term in (3.16) we obtain

$$\int_0^T \sum_{j=1}^N \int_{I_j} (U-\mathbb{P}(U))_x (\mathbb{P}(e))_x dxdt$$

$$\leq \frac{1}{2(1-\gamma)} \int_0^T \sum_{j=1}^N \|(U-\mathbb{P}(U))_x(\cdot,t)\|_{I_j}^2 dt + \frac{(1-\gamma)}{2} \int_0^T \sum_{j=1}^N \|(\mathbb{P}(e))_x(\cdot,t)\|_{I_j}^2 dt.$$

The third term in (3.16) is majored by

$$\int_0^T \left[\sum_{j=1}^N \widehat{(U-\mathbb{P}(U))}_x [\mathbb{P}(e)]\right] dt$$

$$\leq \frac{\Delta x}{2\alpha} \int_0^T \sum_{j=1}^N \{D_x(U-\mathbb{P}(U))\}^2 dt + \frac{\alpha}{2} \int_0^T \sum_{j=1}^N \frac{[\mathbb{P}(e)]^2}{\Delta x} dt.$$

The above three estimates when inserted into (3.16) gives

$$\mathbb{B}(U-\mathbb{P}(U),\mathbb{P}(e)) \leq \frac{1}{2}|||\mathbb{P}(e)(\cdot,T)|||^2 - \frac{1}{2}||\mathbb{P}(e)(\cdot,T)||^2$$

$$+\frac{1}{2(1-\gamma)^2}|||(U-\mathbb{P}(U))(\cdot,T)|||^2 + \frac{\Delta x}{2\alpha}\int_0^T \sum_{j=1}^N \{D_x(U-\mathbb{P}(U))\}^2 dt.$$

This with (3.15) when substituted into (3.16) yields the inequality claimed in Lemma 3.3. ∎

4. Multidimensional diffusion process. In this section, we generalize the DDG method discussed in the previous sections to multiple spatial dimensions $x = (x_1, \ldots, x_d)$. We solve the following diffusion problem:

$$\partial_t U - \sum_{i=1}^d \partial_{x_i}\left(\sum_{j=1}^d a_{ij}(U)\partial_{x_j} U\right) = 0 \quad \text{in} \quad (0,T)\times(0,1)^d, \tag{4.1}$$

$$U(x, t=0) = U_0 \quad \text{on} \quad (0,1)^d, \tag{4.2}$$

with periodic boundary conditions. The diffusion coefficient matrix (a_{ij}) is assumed to be symmetric, semipositive definite.

Notice that the assumption of a unit box geometry and periodic boundary conditions is for simplicity only and is not essential: the method can be designed for arbitrary domain and for nonperiodic boundary conditions.

Let a partition of the unit box $(0,1)^d$ be denoted by shape-regular meshes $T_\Delta = \{K\}$, consisting of a nonoverlapping open element covering completely the unit box. We denote by Δ the piecewise constant mesh function with $\Delta(x) \equiv \Delta_K = diam\{K\}$ when x is in element K. Let each K be a smooth bijective image of a fixed master element: the open hypercube $C = (-1,1)^d$ through $F_K : C \to K$. On C we define spaces of polynomials of degree $k \geq 1$ as follows:

$$P^k = \text{span}\{y^\alpha : \ 0 \leq \alpha_i \leq k, \ 1 \leq i \leq d\}.$$

We denote the finite element space by

$$\mathbb{V}_\Delta = \left\{v : \quad v|_K \circ F_K \in P^k \quad \forall K \in T_\Delta\right\}. \tag{4.3}$$

Note that the master element can also be chosen as the open unit simplex

$$S = \{x \in \mathbb{R}^d : \ 0 < x_1 + \cdots + x_d < 1, \ x_j > 0, j = 1 \cdots d\};$$

then the corresponding polynomial should be changed to $P^k = \text{span}\{y^\alpha : \ 0 \leq |\alpha| \leq k\}$. The DDG method is obtained by discretizing (4.1) directly with the DG method. This is achieved by multiplying the equation by test functions $v \in \mathbb{V}_\Delta$, integrating over an element $K \in T_\Delta$, and integration by parts. We again need to pay special attention to the boundary terms resulting from the procedure of integration by parts, as in the 1D case. Thus we seek piecewise polynomial solution $u \in \mathbb{V}_\Delta$, where $\mathbb{V}_\Delta$ is defined in (4.3) such that for all test functions $v \in \mathbb{V}_\Delta$ we have

$$\int_K u_t v dx + \int_K \sum_{i=1}^d \sum_{j=1}^d a_{ij}(u) \partial_{x_j} u \partial_{x_i} v dx - \int_{\partial K} \widehat{h_{n_K}} v^{int_K} ds = 0, \tag{4.4}$$

where ∂K is the boundary of element K, $n_K = (n_{1,K}, \ldots, n_{d,K})$ is the outward unit normal for element K along the element boundary ∂K, and v^{int_K} denotes the value of v evaluated from inside the element K. Correspondingly, we use v^{ext_K} to denote the value of v evaluated from outside the element K (inside the neighboring element). The numerical flux $\widehat{h_{n_K}}$ is defined similarly to the 1D case as

$$\widehat{h_{n_K}} = \widehat{h_{n_K,K}}\left(u^{int_K}, u^{ext_K}\right) = \sum_{i=1}^d \left(\sum_{j=1}^d \partial_{x_j} \widehat{(b_{ij}(u))} \right) n_i, \tag{4.5}$$

where $b_{ij}(u) = \int^u a_{ij}(s) ds$ and

$$\partial_{x_j} \widehat{(b_{ij}(u))} = \beta_0 \frac{[b_{ij}(u)]}{\Delta} n_j + \overline{\partial_{x_j}(b_{ij}(u))},$$

where locally Δ can be defined as the average of diameters of two neighboring elements sharing one common face. Here we have used the following notations:

$$[u] = u^{ext_K} - u^{int_K} \quad \text{and} \quad \overline{\partial_{x_j} u} = \frac{\partial_{x_j} u^{ext_K} + \partial_{x_j} u^{int_K}}{2}.$$

Note that for hyperrectangle meshes we replace Δ by $\overline{\Delta x_j}$, which denotes the average of lengths of two adjacent elements in the x_j direction only. This way the scheme is consistent with the finite difference scheme when $\beta_0 = 1$. In the general case, the stability is ensured by a larger choice of β_0. The algorithm is now well defined.

We note that the numerical flux defined above enjoys some nice properties similar to those in the 1D case. More precisely, $\widehat{h_{n_K}}(u^{int_K}, u^{ext_K})$ is consistent with $h_{n_K}(u)$ in the sense that $\widehat{h_{n_K}}(u,u) = \sum_{i=1}^{d}(\sum_{j=1}^{d} \partial_{x_j}(b_{ij}(u)))n_i$, which is verified for all u smooth enough. It is also conservative (that is, there is only one flux defined at each face shared by two elements), namely,

$$\widehat{h_{n_K,K}}(a,b) = -\widehat{h_{n_{K'},K'}}(b,a),$$

where K and K' share the same face where the flux is computed and hence $n_{K'} = -n_K$. Moreover, it ensures the L^2-stability of the method.

THEOREM 4.1 (energy stability). *Assume that for $p \in \mathbb{R}$, $\exists\gamma$ and γ^* such that the eigenvalues of matrix $(a_{ij}(p))$ lie between $[\gamma, \gamma^*]$. Consider the DDG scheme with numerical flux chosen in* (4.5). *Then the numerical solution satisfies*

$$\begin{aligned}\int_{(0,1)^d} u^2(x,T)dx + \int_0^T \sum_K \int_K \sum_{i=1}^{d}\sum_{j=1}^{d} a_{ij}(u)u_{x_i}u_{x_j}dxdt \\ + \gamma\beta_0 \int_0^T \sum_K \int_{\partial K} \frac{[u]^2}{\Delta} dsdt \le \int_0^1 U_0^2(x)dx,\end{aligned} \tag{4.6}$$

provided $\beta_0 \ge C(k)(\frac{\gamma^*}{\gamma})^2$ for some $C(k)$, depending on the degree k of the approximating polynomial.

Proof. Setting $v = u$ in (4.4) and summing over all elements, we obtain

$$\frac{d}{dt}\int_\Omega \frac{u^2}{2}dx + \sum_K \int_K \nabla u \cdot (A(u)\nabla u)dx + \sum_K \int_{\partial K} \hat{h}[u]ds = 0, \quad \Omega := [0,1]^d. \tag{4.7}$$

The last term involving the flux (4.5) can be bounded from below as follows:

$$\begin{aligned}\sum_K \int_{\partial K} \hat{h}[u]ds &= \sum_K \int_{\partial K} \left(\frac{\beta_0[u]}{\Delta} \sum_{i,j=1}^{d} n_i \frac{[b_{ij}(u)]}{[u]} n_j + \sum_{i,j=1}^{d} \partial_{x_j} u a_{ij}(u) n_i \right) [u]ds \\ &\ge \frac{\beta_0}{\Delta} \sum_K \int_{\partial K} \gamma[u]^2 ds - \gamma^* \sum_K \int_{\partial K} |\nabla u||n||[u]|ds \\ &\ge \gamma\beta_0 \sum_K \int_{\partial K} \frac{[u]^2}{\Delta} ds - \gamma^* \sum_K \|\nabla u\|_{0,\partial K} \|[u]\|_{0,\partial K} \\ &\ge \frac{\beta_0\gamma}{2} \sum_K \Delta^{-1} \|[u]\|_{0,\partial K}^2 - \frac{(\gamma*)^2}{2\beta_0\gamma} \sum_K \Delta \|\nabla u\|_{0,\partial K}^2,\end{aligned} \tag{4.8}$$

where we used the assumption on matrix $A(u)$, followed by using the inequality $ab \le \epsilon a^2/2 + b^2/(2\epsilon)$ to achieve the last inequality. Using the trace inequality and the fact that $u \in \mathbb{V}_\Delta$ we further obtain

$$\|\nabla u\|_{0,\partial K}^2 \le C\left(\Delta^{-1}\|\nabla u\|_{0,K}^2 + \Delta\|\nabla^2 u\|_{0,K}^2\right) \le C(k)\Delta^{-1}\|\nabla u\|_{0,K}^2.$$

Hence

$$\sum_K \Delta \|\nabla u\|^2_{0,\partial K} \le C(k) \sum_K \|\nabla u\|^2_{0,\partial K} \le \frac{C(k)}{\gamma} \sum_K \int_K \nabla u \cdot (A(u)\nabla u) dx.$$

This, together with (4.8) and when inserted into (4.7), gives

$$\frac{d}{dt}\int_\Omega \frac{u^2}{2} dx + \left(1 - \frac{C(k)}{2\beta_0}\left(\frac{\gamma^*}{\gamma}\right)^2\right) \sum_K \int_K \nabla u \cdot (A(u)\nabla u) dx + \frac{\gamma\beta_0}{2} \sum_K \int_{\partial K} \frac{[u]^2}{\Delta} ds \le 0.$$

Thus the asserted inequality follows from time integration of the above over $[0, T]$ and the fact that $\|u_0\|_{0,\Omega} \le \|U_0\|_{0,\Omega}$, provided $\beta_0 \ge C(k)(\frac{\gamma^*}{\gamma})^2$. ∎

5. Numerical examples. In this section, we provide a few numerical examples to illustrate the accuracy and capacity of the DDG method. We would like to illustrate the high order accuracy of the method through these numerical examples from 1D to 2D linear and nonlinear problems. In particular, we study the numerical performance of the scheme with different admissible numerical fluxes.

Example 5.1 (1D linear diffusion equation).

$$U_t - U_{xx} = 0 \quad \text{in } (0, 2\pi) \tag{5.1}$$

with initial condition $U(x, 0) = \sin(x)$ and periodic boundary conditions. The exact solution is given by $U(x, t) = e^{-t}\sin(x)$. We compute the solution up to $t = 1$. The numerical flux $\widehat{u_x}$ we first test is

$$\widehat{u_x} = D_x(u) = \frac{[u]}{\Delta x} + \overline{u_x}. \tag{5.2}$$

DDG methods based on p^k polynomial approximations with $k = 0, 1, 2$ are tested. We list the L^2 and L^∞ errors in Table 1. Note that in this example and the rest L^∞ error is evaluated on many sample points (200 points per cell). We obtain clean 1st and 2nd order accuracy for p^0 and p^1 approximations. However, we obtain only 2nd order convergence for p^2 approximations.

For higher order polynomial approximations, the proposed numerical flux formula suggests that interface values necessarily involve higher order derivatives of the solution. We first test the scheme (2.12) with $h = \theta\Delta x$, called θ-scheme:

$$D_x u = \frac{7}{12\theta}\frac{[u]}{\Delta x} + \overline{u_x} + \frac{\theta\Delta x}{6}[u_{xx}]. \tag{5.3}$$

In Table 2 we compute p^2 approximations for problem (5.1) with numerical flux (5.3) and list the L^∞ errors and orders with different θ values in interval $(0, 2)$. We would like to point out that almost all θ-schemes give us 2nd order convergence for p^2 polynomial approximations except the one $\theta = 0.5$, i.e., (2.12), which can fully recover the order of 3. Numerically, we observe that the scheme with any fixed β_1 is not sensitive to the coefficient before $\frac{[u]}{\Delta x}$, i.e., β_0, as long as the numerical flux is still admissible.

Numerical results for p^3 approximations with these θ-schemes are displayed in Table 3. Different from the p^2 approximations, all schemes give 4th order convergence. This is in sharp contrast to the p^2 approximations, which give the desired order of 3 only in the case of $\beta_1 = 1/12$.

TABLE 1

Computational domain Ω is $[0, 2\pi]$. L^2 and L^∞ errors at $t = 1.0$. p^k polynomial approximations with $k = 0, 1, 2$. Numerical flux (5.2) is used.

k		N=10	N=20		N=40		N=80	
		error	error	order	error	order	error	order
0	L^2	4.8602E−02	2.3771E−02	1.03	1.1818E−02	1.00	5.9007E−03	1.00
	L^∞	1.1743E−01	5.8023E−02	1.01	2.8923E−02	1.00	1.4450E−02	1.00
1	L^2	1.3400E−02	3.3494E−03	2.00	8.3726E−04	2.00	2.0931E−04	2.00
	L^∞	3.0145E−02	7.5004E−03	2.00	1.8871E−03	2.00	4.7252E−04	2.00
2	L^2	8.5278E−03	2.1377E−03	1.99	5.3476E−04	1.99	1.3371E−04	2.00
	L^∞	1.2082E−02	2.9989E−03	2.00	7.5475E−04	1.99	1.8900E−04	2.00

TABLE 2

L^∞ errors for p^2 approximation with sample θ values in $(0, 2)$ at $t = 1.0$. Numerical flux (5.3) is used.

θ	N=10	N=20		N=40		N=80	
	error	error	order	error	order	error	order
0.1	1.6671E−03	4.1669E−04	2.00	1.0385E−04	2.00	2.5941E−05	2.00
0.3	2.3926E−03	6.2056E−04	1.95	1.5549E−04	2.00	3.8894E−05	2.00
0.49	7.2472E−04	1.0088E−04	2.84	1.6683E−05	2.60	3.4529E−06	2.27
0.5	**7.4892E−04**	**9.1995E−05**	**3.02**	**1.1450E−05**	**3.00**	**1.4296E−06**	**3.00**
0.51	8.1560E−04	1.0970E−04	2.89	1.7893E−05	2.61	3.6315E−06	2.30
0.8	9.8839E−03	2.4693E−03	2.00	6.2113E−04	1.99	1.5553E−04	1.99
1.5	5.6436E−02	1.5037E−02	1.90	3.8570E−03	1.96	9.7047E−04	1.99

TABLE 3

L^∞ errors for p^3 approximation with sample θ values in $(0, 2)$ at $t = 1.0$. Numerical flux (5.3) is used.

θ	N=10	N=20		N=40		N=80	
	error	error	order	error	order	error	order
0.1	8.1835E−05	5.0752E−06	4.01	3.2055E−07	3.98	2.0087E−08	3.99
0.3	5.2604E−05	3.2520E−06	4.01	2.0522E−07	3.98	1.2857E−08	4.00
0.5	1.3097E−04	8.0993E−06	4.01	5.0895E−07	3.99	3.1853E−08	3.99
0.8	5.0961E−04	3.1639E−05	4.00	1.9942E−06	3.98	1.2491E−07	3.99
1.5	2.0972E−03	1.4010E−04	3.90	8.8607E−06	3.98	5.5517E−07	3.99

Next we test the η-scheme with

$$D_x u = \frac{1}{\sqrt{4\pi}} \frac{[u]}{\eta \Delta x} + \overline{u_x} + \frac{\eta \Delta x}{\sqrt{\pi}} [u_{xx}]. \tag{5.4}$$

Similar to the θ-schemes, only $\eta = \frac{\sqrt{\pi}}{12}$ gives fully 3rd order convergence for p^2 polynomial approximations. In Table 4 we list the L^∞ errors with different η values, and the numerical results are comparable to the θ-schemes. Note that careful verification shows that a large class of the θ-schemes and η-schemes satisfy the admissible condition (2.8).

In summary, $\beta_1 = \frac{1}{12}$ numerically gives the optimal $(k+1)$th order of convergence for both p^2 and p^3 polynomial approximations. In Table 5 we use the numerical flux (5.5) to compute the problem with p^2, p^3, and p^4 polynomial approximations:

$$D_x u = \frac{[u]}{\Delta x} + \overline{u_x} + \frac{\Delta x}{12} [u_{xx}]. \tag{5.5}$$

TABLE 4
L^∞ errors and orders comparisons for η-schemes at $t = 1.0$. p^2 polynomial approximation. Numerical flux (5.4) *is used.*

η	N=10	N=20		N=40		N=80	
	error	error	order	error	order	error	order
0.05	1.4733E−03	3.5730E−04	2.04	8.8847E−05	2.00	2.2182E−05	2.00
0.1	1.4427E−03	3.4865E−04	2.05	8.6743E−05	2.00	2.1660E−05	2.00
$\frac{\sqrt{\pi}}{12}$	**7.3278E−04**	**9.1488E−05**	**3.00**	**1.1434E−05**	**3.00**	**1.4291E−06**	**3.00**
0.2	3.1376E−03	7.7015E−04	2.02	1.9054E−04	2.01	4.7511E−05	2.00
0.5	4.7350E−02	1.2413E−02	1.93	3.1738E−03	1.96	7.9794E−04	1.99

TABLE 5
L^2 and L^∞ errors at $t = 1.0$ with p^k approximations $k = 2, 3, 4$. Numerical flux (5.5) *is used.*

k		N=10	N=20		N=40		N=80	
		error	error	order	error	order	error	order
2	L^2	3.9238E−04	4.7037E−05	3.06	5.8181E−06	3.01	7.2535E−07	3.00
	L^∞	7.5595E−04	9.2213E−05	3.03	1.1456E−05	3.00	1.4298E−06	3.00
3	L^2	9.2531E−05	5.7809E−06	4.00	3.6128E−07	4.00	2.2579E−08	4.00
	L^∞	1.5351E−04	9.5014E−06	4.01	5.9750E−07	3.99	3.7403E−08	3.99
4	L^2	5.5070E−05	3.4999E−06	3.97	2.1965E−07	3.99	1.3743E−08	3.99
	L^∞	7.7911E−05	4.8932E−06	3.99	3.0974E−07	3.98	1.9422E−08	3.99

Similar to the p^2 case we lose one order of accuracy for p^4 approximations. These results together indicate that for even-order, $k = 2m$, polynomial approximations, the coefficient β_m seems indispensable.

Example 5.2 (1D linear diffusion equation with higher order polynomial approximations). We study the same problem as the one in Example 5.1 with the numerical flux chosen as

$$D_x u = 2\frac{[u]}{\Delta x} + \overline{u_x}. \tag{5.6}$$

As discussed in section 2, we find out that with β_0 (the coefficient before $[u]/\Delta x$) big enough the numerical flux formula (2.7) with the first two terms is admissible. In this example, we test the DDG scheme with higher order polynomial approximations $p^k, k = 2, 3, 4, 5, 6, 7$. Errors and orders are listed in Table 6. We obtain kth order accuracy for even k and $(k+1)$th order accuracy for odd k.

Example 5.3 (1D linear diffusion equation with nonuniform mesh). Again, we study the same problem as the one in Example 5.1. Here the partition of the domain $[0, 2\pi]$ consists of a repeated pattern of $1.1\Delta x$ and $0.9\Delta x$ for odd and even numbers of indexes $i = 1, \ldots, N$, respectively, where $\Delta x = 2\pi/N$ with even number N. The numerical flux we use is

$$D_x(u) = \frac{[u]}{\Delta x} + \overline{u_x}.$$

We obtain similar results as Example 5.2. Errors and orders are listed in Table 7.

Example 5.4 (1D nonlinear diffusion equations).

$$U_t - (2UU_x)_x = 0 \quad \text{in } [-12, 12]. \tag{5.7}$$

TABLE 6
High order polynomial approximations $(p^k, k = 2, 3, 4, 5, 6, 7)$ *with numerical flux* (5.6). L^2 *and* L^∞ *errors at* $t = 1.0$.

k		N=4	N=8		N=12		N=16	
		error	error	order	error	order	error	order
2	L^2	2.3913E−02	6.5160E−03	1.88	2.9383E−03	1.96	1.6610E−03	1.98
	L^∞	3.0219E−02	8.9725E−03	1.75	4.1066E−03	1.93	2.3335E−03	1.96
3	L^2	3.6671E−03	2.2958E−04	4.00	4.5459E−05	3.99	1.4397E−05	4.00
	L^∞	4.5777E−03	3.6566E−04	3.65	7.5484E−05	3.90	2.4253E−05	3.95
4	L^2	3.5708E−04	2.7557E−05	3.60	5.6506E−06	3.91	1.8111E−06	3.96
	L^∞	4.4087E−04	3.7131E−05	3.70	7.8142E−06	3.85	2.5288E−06	3.93
5	L^2	3.9466E−05	6.4001E−07	5.95	5.6637E−08	5.98	1.0109E−08	5.99
	L^∞	4.6456E−05	9.2965E−07	5.64	8.4966E−08	5.90	1.5332E−08	5.95
6	L^2	1.8891E−06	4.1364E−08	5.51	3.8499E−09	5.86	6.9911E−10	5.93
	L^∞	2.4795E−06	5.5327E−08	5.49	5.3001E−09	5.78	9.7347E−10	5.89
7	L^2	2.1144E−07	8.9249E−10	7.89	3.5478E−11	7.95	3.6571E−12	7.90
	L^∞	2.5297E−07	1.2566E−09	7.65	5.1137E−11	7.90	5.3087E−12	7.87

TABLE 7
Nonuniform mesh test. L^2 *and* L^∞ *errors at* $t = 1.0$ *with* $k = 0, 1, 2, 3, 4, 5$.

k		N=10	N=20		N=40		N=80	
		error	error	order	error	order	error	order
0	L^2	4.8970E−02	2.3903E−02	1.03	1.1879E−02	1.00	5.9304E−03	1.00
	L^∞	1.3021E−01	6.3933E−02	1.02	3.1828E−02	1.00	1.5897E−02	1.00
1	L^2	1.4011E−02	3.4807E−03	2.00	8.6898E−04	2.00	2.1717E−04	2.00
	L^∞	3.8329E−02	9.3268E−03	2.00	2.3522E−03	1.99	5.8881E−04	2.00
2	L^2	8.7915E−03	2.2023E−03	1.99	5.5083E−04	2.00	1.3772E−04	2.00
	L^∞	1.2720E−02	3.0946E−03	2.03	7.7858E−04	1.99	1.9475E−04	2.00
3	L^2	1.9041E−04	1.1836E−05	4.00	7.3911E−07	4.00	4.6186E−08	4.00
	L^∞	3.9591E−04	2.4148E−05	4.03	1.5144E−06	4.00	9.4854E−08	4.00
4	L^2	2.5721E−05	1.6464E−06	3.97	1.0353E−07	4.00	6.4802E−09	4.00
	L^∞	3.7917E−05	2.3196E−06	4.03	1.4645E−07	3.99	9.1649E−09	4.00
5	L^2	3.7561E−07	5.8458E−09	6.00	9.1163E−11	6.00	1.4244E−12	6.00
	L^∞	7.5379E−07	1.1563E−08	6.02	1.8114E−10	6.00	2.8303E−12	6.00

The Barenblatt's solution with compact support is given as

$$U(x,t) = \begin{cases} (t+1)^{-\frac{1}{3}} \left(6 - \frac{x^2}{12(t+1)^{\frac{2}{3}}}\right), & |x| < 6(t+1)^{\frac{1}{3}}, \\ 0, & |x| \geq 6(t+1)^{\frac{1}{3}}. \end{cases} \tag{5.8}$$

We take the following numerical flux for this nonlinear problem:

$$\widehat{2uu_x} = \widehat{(u^2)_x} = \frac{[u^2]}{\Delta x} + \overline{(u^2)_x} + \frac{\Delta x}{12}\left[(u^2)_{xx}\right].$$

Both L^2 and L^∞ errors at $t = 1$ are evaluated in domain $[-6, 6]$ where the solution is smooth. Accuracy data are listed in Table 8. We have $(k+1)$th order accuracy with p^k polynomial approximations. Propagation of the compact wave using both P^1 and P^2 elements is plotted in Figure 2. A zoomed-in figure of the left corner at $t = 4$ is plotted in Figure 3. The DDG scheme can sharply capture the contacts with discontinuous derivatives.

TABLE 8

Computational domain Ω *is* $[-12, 12]$. *Exact solution is given as* (5.8). L^2 *and* L^∞ *errors are computed in smooth region* $[-6, 6]$ *with* $k = 0, 1, 2$ *at* $t = 1.0$.

k		N=40	N=80		N=160		N=320	
		error	error	order	error	order	error	order
0	L^2	3.8184E−02	1.8363E−02	1.05	9.0076E−03	1.03	4.4613E−03	1.01
	L^∞	1.5739E−01	7.7079E−02	1.03	3.8038E−02	1.02	1.8887E−02	1.01
1	L^2	2.8239E−03	6.8004E−04	2.05	1.6478E−04	2.04	4.0616E−05	2.02
	L^∞	1.3169E−02	2.8427E−03	2.21	6.6660E−04	2.09	1.6294E−04	2.03
2	L^2	2.2321E−04	1.2519E−05	4.15	1.1960E−06	3.38	1.4516E−07	3.04
	L^∞	2.5443E−03	8.3480E−05	4.92	3.7225E−06	4.48	4.1537E−07	3.16

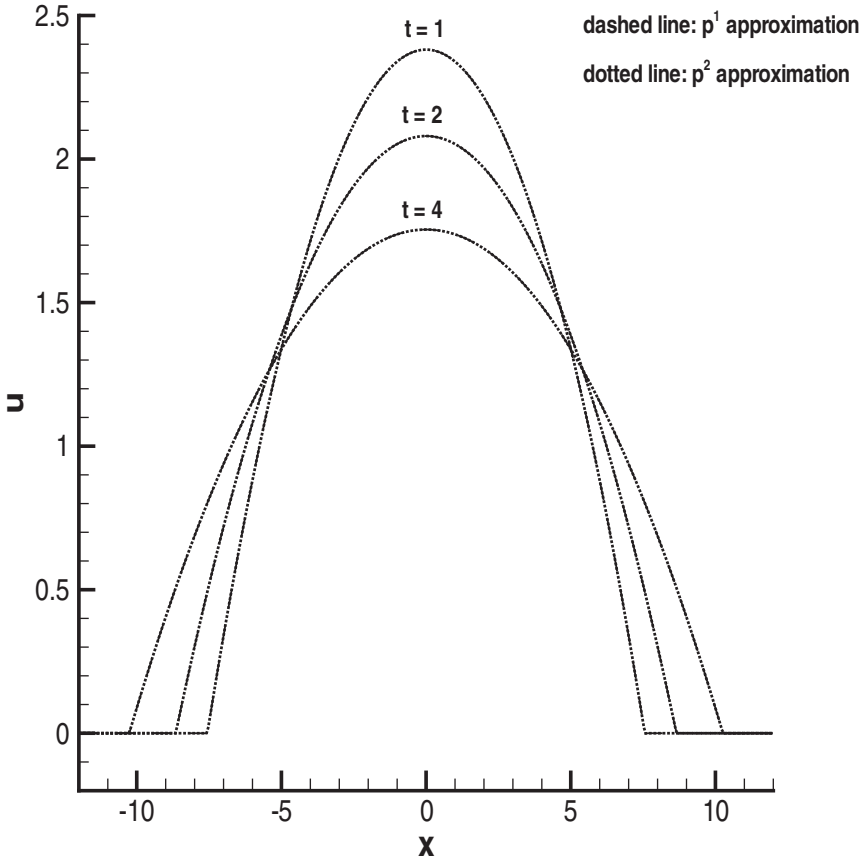

FIG. 2. *Nonlinear diffusion equation* (5.7). *Piecewise linear* (p^1) *and piecewise quadratic* (p^2) *approximations with mesh* $N = 400$.

Example 5.5 (2D linear diffusion equation).

$$U_t - (U_{xx} + U_{yy}) = 0 \quad \text{in } (0, 2\pi) \times (0, 2\pi) \tag{5.9}$$

with initial condition $U(x, y, 0) = \sin(x + y)$ and periodic boundary conditions. The exact solution is $U(x, y, t) = e^{-2t}\sin(x + y)$. We compute the solution up to $t = 1$ on the uniform rectangular mesh $I_{ij} = I_i \times I_j$. L^2 and L^∞ errors are listed in Table 9. $k + 1$ orders of convergence are obtained for p^k elements with $k \leq 3$.

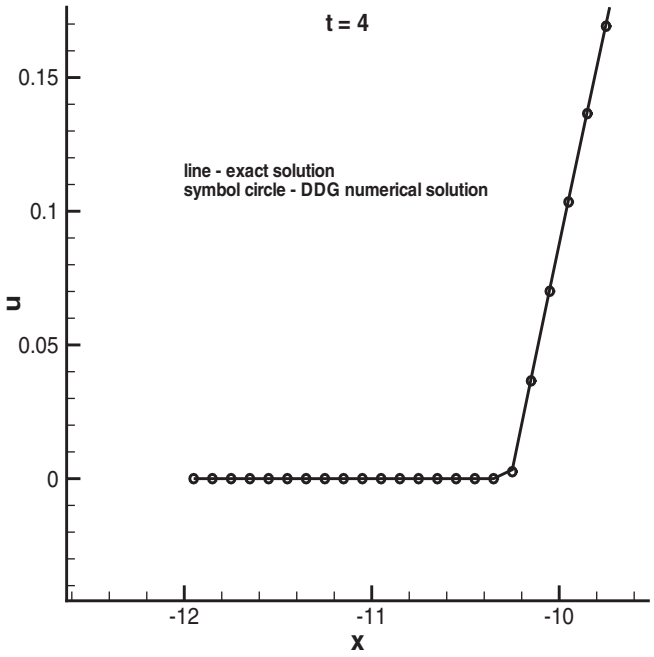

FIG. 3. *Zoom in of Figure* 2 *at* $t = 4$ *at the left corner where the solution has a discontinuous derivative.*

TABLE 9

Computational domain Ω *is* $[0, 2\pi] \times [0, 2\pi]$ *with rectangular mesh* $N \times N$. L^2 *and* L^∞ *errors at* $t = 1.0$. p^k *polynomials with* $k = 0, 1, 2, 3, 4$. *Numerical flux* (5.10) *is used.*

k		N=10	N=20		N=40		N=80	
		error	error	order	error	order	error	order
0	L^2	2.5993E−02	1.2456E−02	1.06	6.1599E−03	1.01	3.0714E−03	1.00
	L^∞	8.4874E−02	4.2512E−02	1.00	2.1258E−02	1.00	1.0629E−02	1.00
1	L^2	1.0305E−02	2.5482E−03	2.01	6.3522E−04	2.00	1.5869E−04	2.00
	L^∞	3.5728E−02	8.8609E−03	2.01	2.2234E−03	1.99	5.5637E−04	2.00
2	L^2	1.1503E−03	1.4042E−04	3.03	1.7437E−05	3.00	2.1759E−06	3.00
	L^∞	7.4943E−03	9.4734E−04	2.98	1.1872E−04	2.99	1.4849E−05	3.00
3	L^2	1.0777E−04	6.2265E−06	4.11	3.8168E−07	4.02	2.3740E−08	4.00
	L^∞	5.7940E−04	3.8896E−05	3.89	2.4742E−06	3.97	1.5531E−07	3.99
4	L^2	5.3967E−05	3.4795E−06	3.95	2.1932E−07	3.98	1.3738E−08	3.99
	L^∞	8.5886E−05	5.1584E−06	4.05	3.1396E−07	4.03	1.9488E−08	4.01

The DDG scheme in 2D with rectangular mesh is a straightforward extension of the 1D scheme. The numerical flux $\widehat{u_x}$ at $x_{i+1/2}$ used in this example is defined as follows:

$$\widehat{u_x}|_{x_{i+1/2}} = \frac{[u]}{\Delta x} + \overline{u_x} + \frac{\Delta x}{12}[u_{xx}]. \tag{5.10}$$

Flux $\widehat{u_y}$ at $y_{j+1/2}$ is defined in a similar fashion.

6. Concluding remarks. We have proposed a new DG finite element method for solving diffusion problems. The scheme is formulated using the direct weak for-

mulation for parabolic equations, combined with a careful design of interface values of the solution derivative. Unlike the traditional LDG method, the method in this paper is applied without introducing any auxiliary variables or rewriting the original equation into a 1st order system. The proposed numerical flux formula for solution derivatives is consistent and conservative. A concept of admissibility is further introduced to identify a class of numerical fluxes so that the nonlinear stability for both 1D and multidimensional problems are ensured. For the 1D linear case, kth order accuracy in an energy norm is proven when using kth degree polynomials. A series of numerical examples are presented to demonstrate the high order accuracy of the method and its capacity to sharply capture solutions with discontinuous derivatives. In particular, the optimal $(k+1)$th order accuracy is attained for $k = 0, 1, 2, 3$. The method maintains the usual features of DG methods such as high order accuracy and easiness to handle complicated geometry. Moreover, our DDG method has an advantage of easier formulation and implementation and efficient computation of solutions. The compactness of the scheme allows efficient parallelization and hp-adaptivity.

The numerical tests show the strong dependence of the order of convergence of the DDG method on the choice of numerical fluxes. The development of even higher order DDG methods with further analysis of optimal choices for β_i, $i \geq 1$, will be studied in a future work. The DDG method for convection-diffusion problems can be defined by applying the procedure described above for the diffusion term combined with numerical fluxes for the convection term developed previously for hyperbolic conservation laws.

7. Appendix. The Bramble–Hilbert lemma. Let Ω be a simply connected Lipschitz domain in R^d and $(m, k) \in \mathbb{N}^2$, $p, q \in [1, \infty]$. If (p, q, k, m) satisfies

$$\frac{1}{q} > \frac{1}{p} - \frac{k+1-m}{d}, \tag{7.1}$$

then the Sobolev space $W^{k+1,p}(\Omega)$ is continuously embedded into $W^{m,q}(\Omega)$. With this setting we recall the celebrated Bramble–Hilbert lemma.

LEMMA 7.1 (Bramble–Hilbert). *Let l be a linear operator mapping $W^{k+1,p}(\Omega)$ into $W^{m,q}(\Omega)$ and $P_k(\Omega) \subset Ker(l)$. Then there exists a constant $C(\Omega) > 0$ such that for all $u \in W^{m,q}(\Omega)$*

$$|l(v)|_{m,q} \leq C|v|_{k+1,p}. \tag{7.2}$$

The assumption $P_k(\Omega) \subset Ker(l)$ is used to ensure the Sobolev quotient norm in $W^{k+1,p}/P_k(\Omega)$ to be equivalent to the Sobolev seminorm in $W^{k+1,p}(\Omega)$.

In the 1D case I_j is affine equivalent to $\omega = [0, 1]$ through a linear mapping

$$x = x_{j-1/2} + \xi \Delta x, \quad \xi \in [0, 1].$$

Taking $l = I - \mathbb{P}$ and using the scaling argument, the Bramble–Hilbert lemma enables us to obtain

$$|v - \mathbb{P}v|_{m,q,I_j} \leq C(\Delta x)^{k+1-m+\frac{1}{q}-\frac{1}{p}} |v|_{k+1,p,I_j}, \quad v \in W^{k+1,p}(I_j).$$

Here C depends only on $[0, 1]$ and the projection operator $\mathbb{P}$ but is independent of Δx. Estimates in Lemma 3.1 are immediate if
(i) we set $p = q = 2$ and assume $m < k + 1$;
(ii) we set $q = \infty$ and $p = 2$ and assume $m < k + 1/2$.

Solution gradient for the heat equation. Consider the heat equation $u_t = u_{xx}$ with smooth initial data g, having only one discontinuity at $x = 0$. A straightforward calculation from the solution formula

$$u(x,t) = \frac{1}{\sqrt{4\pi t}} \int_{-\infty}^{\infty} e^{-(x-y)^2/(4t)} g(y) dy$$

gives

$$\begin{aligned} u_x(0,t) &= \sum \frac{2^{m-1}}{(2m-1)!!} t^m \left[\partial_x^{2m} g\right] / \sqrt{\pi t} + \sum \frac{2^m}{(2m)!!} t^m \overline{\partial_x^{2m+1} g} \\ &= \frac{1}{\sqrt{4\pi t}} [g] + \overline{\partial_x g} + \sqrt{\frac{t}{\pi}} \left[\partial_x^2 g\right] + t \overline{\partial_x^3 g} + \cdots, \end{aligned} \tag{7.3}$$

where the jump or the average of g and its derivatives are involved to evaluate u_x at $x = 0$.

Acknowledgments. The authors thank the anonymous referees who provided valuable comments resulting in improvements in this paper.

REFERENCES

[1] D. N. Arnold, F. Brezzi, B. Cockburn, and L. D. Marini, *Unified analysis of discontinuous Galerkin methods for elliptic problems*, SIAM J. Numer. Anal., 39 (2002), pp. 1749–1779.
[2] D. N. Arnold, *An interior penalty finite element method with discontinuous elements*, SIAM J. Numer. Anal., 19 (1982), pp. 742–760.
[3] G. A. Baker, *Finite element methods for elliptic equations using nonconforming elements*, Math. Comp., 31 (1977), pp. 45–59.
[4] F. Bassi and S. Rebay, *A high-order accurate discontinuous finite element method for the numerical solution of the compressible Navier-Stokes equations*, J. Comput. Phys., 131 (1997), pp. 267–279.
[5] C. E. Baumann and J. T. Oden, *A discontinuous hp finite element method for convection-diffusion problems*, Comput. Methods Appl. Mech. Engrg., 175 (1999), pp. 311–341.
[6] F. Brezzi, B. Cockburn, L. D. Marini, and E. Süli, *Stabilization mechanisms in discontinuous Galerkin finite element methods*, Comput. Methods Appl. Mech. Engrg., 195 (2006), pp. 3293–3310.
[7] P. Castillo, B. Cockburn, I. Perugia, and D. Schötzau, *An a priori error analysis of the local discontinuous Galerkin method for elliptic problems*, SIAM J. Numer. Anal., 38 (2000), pp. 1676–1706.
[8] F. Celiker and B. Cockburn, *Superconvergence of the numerical traces of discontinuous Galerkin and hybridized methods for convection-diffusion problems in one space dimension*, Math. Comp., 76 (2007), pp. 67–96.
[9] Y. Cheng and C.-W. Shu, *A discontinuous Galerkin finite element method for time dependent partial differential equations with higher order derivatives*, Math. Comp., 77 (2008), pp. 699–730.
[10] B. Cockburn and C. Dawson, *Approximation of the velocity by coupling discontinuous Galerkin and mixed finite element methods for flow problems*, Comput. Geosci., 6 (2002), pp. 505–522.
[11] B. Cockburn, S. Hou, and C.-W. Shu, *The Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws.* IV. *The multidimensional case*, Math. Comp., 54 (1990), pp. 545–581.
[12] B. Cockburn, G. Kanschat, and D. Schotzau, *A locally conservative LDG method for the incompressible Navier-Stokes equations*, Math. Comp., 74 (2005), pp. 1067–1095.
[13] B. Cockburn, G. E. Karniadakis, and C.-W. Shu, *The development of discontinuous Galerkin methods*, in Discontinuous Galerkin Methods, Lect. Notes Comput. Sci. Eng. 11, Springer, Berlin, 2000, pp. 3–50.
[14] 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., 84 (1989), pp. 90–113.

[15] B. Cockburn and C.-W. Shu, *TVB Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws.* II. *General framework*, Math. Comp., 52 (1989), pp. 411–435.
[16] B. Cockburn and C.-W. Shu, *The Runge-Kutta local projection P^1-discontinuous-Galerkin finite element method for scalar conservation laws*, RAIRO Modél. Math. Anal. Numér., 25 (1991), pp. 337–361.
[17] B. Cockburn and C.-W. Shu, *The local discontinuous Galerkin method for time-dependent convection-diffusion systems*, SIAM J. Numer. Anal., 35 (1998), pp. 2440–2463.
[18] B. Cockburn and C.-W. Shu, *The Runge-Kutta discontinuous Galerkin method for conservation laws.* V. *Multidimensional systems*, J. Comput. Phys., 141 (1998), pp. 199–224.
[19] B. Cockburn and C.-W. Shu, *Runge-Kutta discontinuous Galerkin methods for convection-dominated problems*, J. Sci. Comput., 16 (2001), pp. 173–261.
[20] B. Cockburn, *Discontinuous Galerkin methods for convection-dominated problems*, in High-order Methods for Computational Physics, Lect. Notes Comput. Sci. Eng. 9, Springer, Berlin, 1999, pp. 69–224.
[21] G. Gassner, F. Lörcher, and C.-D. Munz, *A contribution to the construction of diffusion fluxes for finite volume and discontinuous Galerkin schemes*, J. Comput. Phys., 224 (2007), pp. 1049–1063.
[22] D. Levy, C.-W. Shu, and J. Yan, *Local discontinuous Galerkin methods for nonlinear dispersive equations*, J. Comput. Phys., 196 (2004), pp. 751–772.
[23] H. Liu and J. Yan, *A local discontinuous Galerkin method for the Korteweg-de Vries equation with boundary effect*, J. Comput. Phys., 215 (2006), pp. 197–218.
[24] J. T. Oden, I. Babuška, and C. E. Baumann, *A discontinuous hp finite element method for diffusion problems*, J. Comput. Phys., 146 (1998), pp. 491–519.
[25] W. H. Reed and T. R. Hill, *Triangular Mesh Methods for the Neutron Transport Equation*, Technical report LA-UR-73-479, Los Alamos Scientific Laboratory, Los Alamos, NM, 1973.
[26] C.-W. Shu and S. Osher, *Efficient implementation of essentially nonoscillatory shock-capturing schemes*, J. Comput. Phys., 77 (1988), pp. 439–471.
[27] C.-W. Shu and S. Osher, *Efficient implementation of essentially nonoscillatory shock-capturing schemes.* II, J. Comput. Phys., 83 (1989), pp. 32–78.
[28] C.-W. Shu, *Different formulations of the discontinuous Galerkin method for the viscous terms*, Advances in Scientific Computing, Z.-C. Shi, M. Mu, W. Xue, and J. Zou, eds., Science Press, Beijing, China, 2001, pp. 144–155.
[29] B. van Leer and S. Nomura, *Discontinuous Galerkin for diffusion*, in Proceedings of the 17th AIAA Computational Fluid Dynamics Conference, Toronto, Canada, 2005, AIAA-2005-5108.
[30] M. F. Wheeler, *An elliptic collocation-finite element method with interior penalties*, SIAM J. Numer. Anal., 15 (1978), pp. 152–161.
[31] Y. Xu and C.-W. Shu, *Local discontinuous Galerkin methods for nonlinear Schrödinger equations*, J. Comput. Phys., 205 (2005), pp. 72–97.
[32] J. Yan and C.-W. Shu, *A local discontinuous Galerkin method for KdV type equations*, SIAM J. Numer. Anal., 40 (2002), pp. 769–791.