Journal of Scientific Computing, Vol. 16, No. 3, September 2001 (© 2001)

Review Article

Runge–Kutta Discontinuous Galerkin Methods for Convection-Dominated Problems

Bernardo Cockburn[1] **and Chi-Wang Shu**[2]

Received April 8, 2001; accepted August 27, 2001

In this paper, we review the development of the Runge–Kutta discontinuous Galerkin (RKDG) methods for non-linear convection-dominated problems. These robust and accurate methods have made their way into the main stream of computational fluid dynamics and are quickly finding use in a wide variety of applications. They combine a special class of Runge–Kutta time discretizations, that allows the method to be non-linearly stable regardless of its accuracy, with a finite element space discretization by discontinuous approximations, that incorporates the ideas of *numerical fluxes* and *slope limiters* coined during the remarkable development of the *high-resolution* finite difference and finite volume schemes. The resulting RKDG methods are stable, high-order accurate, and highly parallelizable schemes that can easily handle complicated geometries and boundary conditions. We review the theoretical and algorithmic aspects of these methods and show several applications including nonlinear conservation laws, the compressible and incompressible Navier–Stokes equations, and Hamilton–Jacobi-like equations.

KEY WORDS: Discontinuous Galerkin methods; non-linear conservation laws; convection-diffusion equations.

1. INTRODUCTION

In this paper, we review the development of the Runge–Kutta discontinuous Galerkin (RKDG) methods for non-linear convection-dominated problems. These are methods that have recently found their way into the main stream of

[1] School of Mathematics, University of Minnesota, Minneapolis, Minnesota 55455. E-mail: cockburn@math.umn.edu

[2] Division of Applied Mathematics, Brown University, Providence, Rhode Island 02912. E-mail: shu@cfm.brown.edu

0885-7474/01/0900-0173/0 © 2001 Plenum Publishing Corporation

computational fluid dynamics and are currently being applied to a variety of situations including problems for which they were not originally intended, like purely elliptic systems.

Numerical methods for convection-dominated problems are very useful in a wide variety of applications. Indeed, as pointed out in [44]: "Practical problems in which *convection* plays an important role arise in applications as diverse as meteorology, weather-forecasting, oceanography, gas dynamics, turbomachinery, turbulent flows, granular flows, oil recovery simulation, modeling of shallow waters, transport of contaminant in porous media, visco-elastic flows, semiconductor device simulation, magneto-hydrodynamics, and electro-magnetism, among many others. This is why devising robust, accurate and efficient methods for numerically solving these problems is of considerable importance."

The need for such methods prompted and sustained the remarkable development of the so-called *high-resolution* finite difference and finite volume methods for non-linear hyperbolic systems in divergence form:

$$u_t + \nabla \cdot f(u) = 0$$

see, for example, the monograph by LeVeque [84]. The satisfactory approximation of the exact solutions of these systems is particularly difficult because of the presence of discontinuities in the exact solution. Let us describe this difficulty in the scalar case. On the one hand, the physically relevant solution, called the *entropy solution*, can be captured by means of the so-called *monotone* schemes; unfortunately, they are only first-order accurate when the solution is smooth and display a poor approximation of moving discontinuities. On the other hand, high-order accurate schemes generate spurious oscillations around the discontinuities which, due to the non-linear nature of the equation, can also induce the convergence of the method to a solution that is *not* the entropy solution; see the 1976 paper by Harten *et al.* [69].

This impasse between high-order accuracy and convergence to the entropy solution was solved by the high-resolution schemes. The success of these methods is mainly due to two facts. First, the non-linear conservation laws are enforced locally; that is, the averages of the approximation u_h on each element or cell K, are evolved in time by imposing that

$$\int_K (u_h)_t + \int_{\partial K} \hat{f}(u_h) \cdot n_K \, ds = 0$$

where n_K denotes the outward unit normal to ∂K and $\hat{f}(u_h)$ is the so-called *approximate Riemann solver* or *numerical flux*. We can see that the numerical flux $\hat{f}(u_h)$ is an approximation to the value of $f(u)$ on the boundary of

the elements K. It is devised in such a way that, when u is a scalar-valued function and u_h is piecewise-constant, the resulting method is a monotone scheme; as we pointed out, it is always stable and converges to the exact solution. Second, when the approximate solution u_h is not piecewise-constant, the stability of the method does not follow from the form of the numerical fluxes anymore and has to be enforced by means of *flux* or *slope limiters*. Indeed, once the averages have been evolved in time, the remaining degrees of freedom of u_h are usually determined from them by means of a *reconstruction* step whose main objective is to achieve high-order accuracy; the flux or slope limiters are then applied in order to render the method stable while maintaining its high-order accuracy.

However, these methods cannot handle complex geometries and boundary conditions and achieve high-order accuracy as easily as finite element methods do. On the other hand, most finite element methods for non-linear conservation laws do not enforce the conservation law locally, a property highly valued in practice, and do not satisfy maximum principles (or other stability properties like total variation boundedness) which are essential in many practical situations. More importantly, they give rise to systems of equations that have to be solved implicitly which renders them quite inefficient when strong shocks are present; see the analysis of this fact for non-linear scalar hyperbolic conservation laws by Bourgeat and Cockburn [27].

1.1. The RKDG Method for Purely Convective Non-Linear Problems

The RKDG methods, introduced and studied by the authors and their collaborators [49, 48, 46, 41, 51], realize a fortunate compromise between these two types of numerical schemes by incorporating the ideas of numerical fluxes and slope limiters into a finite element framework. Next, we give a brief idea of how to construct the RKDG methods. We proceed in three steps:

Step 1: The DG Space Discretization. First, the conservation law is discretized in space by using a discontinuous Galerkin (DG) method. A discontinuous approximate solution u_h is sought such that when restricted to the element K, it belongs to the finite dimensional space $\mathscr{U}(K)$, typically a space of polynomials. It is defined by imposing that, for all $v_h \in \mathscr{U}(K)$,

$$\int_K (u_h)_t \, v_h \, dx - \int_K f(u_h) \cdot \nabla v_h \, dx + \int_{\partial K} \hat{f}(u_h) \cdot n_K v_h \, ds = 0$$

Note that it is here that the notion of approximate Riemann solver or numerical flux is actually incorporated into the method. Like all finite element methods, complex geometries and boundary conditions are very easily dealt with and high-order accuracy can be easily obtained. Moreover, since the approximation is discontinuous, the so-called *mass matrix* is block diagonal and hence, easily invertible.

Step 2: The RK Time Discretization. Then, we discretize the resulting system of ordinary differential equations, $\frac{d}{dt} u_h = L(u_h)$, by using the following *explicit* high-order accurate Runge–Kutta (RK) methods:

1. Set $u_h^{(0)} = u_h^n$;
2. For $i = 1,\ldots, \mathscr{K}$ compute the intermediate functions:

$$u_h^{(i)} = \sum_{l=0}^{i-1} \alpha_{il} w_h^{il}, \qquad w_h^{il} = u_h^{(l)} + \frac{\beta_{il}}{\alpha_{il}} \Delta t^n L_h(u_h^{(l)})$$

3. Set $u_h^{n+1} = u_h^{\mathscr{K}}$.

The distinctive feature of these RK methods is that their stability follows from the stability of the mapping $u_h^{(l)} \mapsto w_h^{il}$ defining the intermediate steps. More precisely, if for some semi-norm $|\cdot|$, we have that $|w_h^{il}| \leqslant |u_h^{(l)}|$, then we have $|u_h^{n+1}| \leqslant |u_h^n|$.

Step 3: The Generalized Slope Limiter. Finally, a *generalized slope limiter* $\Lambda\Pi_h$, which is a *non-linear* projection operator, is devised in such a way that if $u_h^{(l)} = \Lambda\Pi_h v_h$ for some function v_h, then the mapping $u_h^{(l)} \mapsto w_h^{il}$ is stable, that is, $|w_h^{il}| \leqslant |u_h^{(l)}|$.

The above time-marching algorithm is then modified as follows:

1. Set $u_h^{(0)} = u_h^n$;
2. For $i = 1,\ldots, \mathscr{K}$ compute the intermediate functions:

$$u_h^{(i)} = \Lambda\Pi_h \left(\sum_{l=0}^{i-1} \alpha_{il} w_h^{il} \right), \qquad w_h^{il} = u_h^{(l)} + \frac{\beta_{il}}{\alpha_{il}} \Delta t^n L_h(u_h^{(l)})$$

3. Set $u_h^{n+1} = u_h^{\mathscr{K}}$.

This is the general form of the RKDG methods; they can be proven to be stable in the semi-norm $|\cdot|$, that is, that $|u_h^n| \leqslant |u_h^0| \leqslant C\,|u_0|$, if the approximation to the initial data, u_h^0 is chosen in a reasonable way, of course.

Note that the RKDG method is devised in such a way that when piecewise constant approximations are used for the space discretization and the forward Euler method is employed for the time discretization, a standard finite volume scheme is obtained; in this case, the generalized slope

limiter $\Lambda\Pi_h$ is nothing but the identity. Thus, the RKDG methods can be considered to be a generalization of finite volume methods. When high-order degree polynomials are used, a high-order RK method that matches the accuracy of the space discretization has to be used which renders the resulting method high-order accurate. In such a case, the use of the generalized slope limiters $\Lambda\Pi_h$ is *crucial* to ensure the stability of the method; indeed, although the use of the generalized slope limiter turns out to be unnecessary when the solution is smooth, it is indispensable when strong shocks are present. The fact that it is possible to construct generalized slope limiters $\Lambda\Pi_h$ that enforce stability *without* degrading the high-order accuracy achieved by the space and time discretizations is one of the most remarkable features of the construction of the RKDG methods.

Note also the high degree of *locality* that the RKDG methods display. First, thanks to the structure of the DG space discretization *and* to the explicitness of the RK time discretization, to update the degrees of freedom inside an element K, only the information about the elements sharing edges with K is used in each inner RK step. This property is not shared by any of the above mentioned high-resolution methods, which in their reconstruction step typically use the information associated to far-away elements. Finally, let us point out that to compute $\Lambda\Pi_h(u_h)$ in the element K, only information about u_h on elements sharing edges with K is necessary. These properties render the RKDG methods highly parallelizable.

Let us briefly illustrate some of the main features of the RKDG methods:

• **Capturing shocks**. First, let us show in a simple example that the RKDG methods can capture shocks as well as any high-resolution finite difference or finite volume scheme does [38]. Consider the approximation of the entropy solution of the inviscid Burgers equation

$$u_t + (u^2/2)_x = 0$$

on the domain $(0, 1) \times (0, T)$ with initial condition $1/4 + \sin(\pi(2x-1))/2$ and periodic boundary conditions. In Fig. 1.1, we display the RKDG solution using piecewise linear and piecewise quadratic approximations; note how, in both cases, the shock has been captured within three elements as would be expected of any high-resolution scheme.

• **The artificial dissipation and the order of accuracy**. Let us now illustrate the relation between the *dissipation* of the RKDG methods and their order of accuracy. Consider the one and two dimensional transport equation

$$u_t + u_x = 0, \qquad \text{or} \qquad u_t + u_x + u_y = 0$$

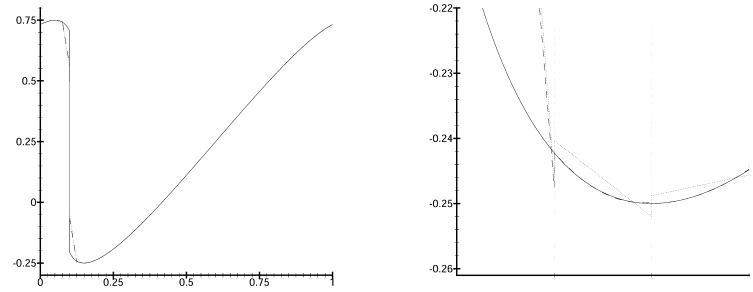

Fig. 1.1. Burgers equation: Comparison of the exact and the RKDG solutions obtained with $\Delta x = 1/40$ at $T = 0.40$. Full domain (left) and zoom on three elements (right) the first of which contains the exact shock. Exact solution (solid line), piecewise linear approximation (dotted line), and piecewise quadratic approximation (dashed line).

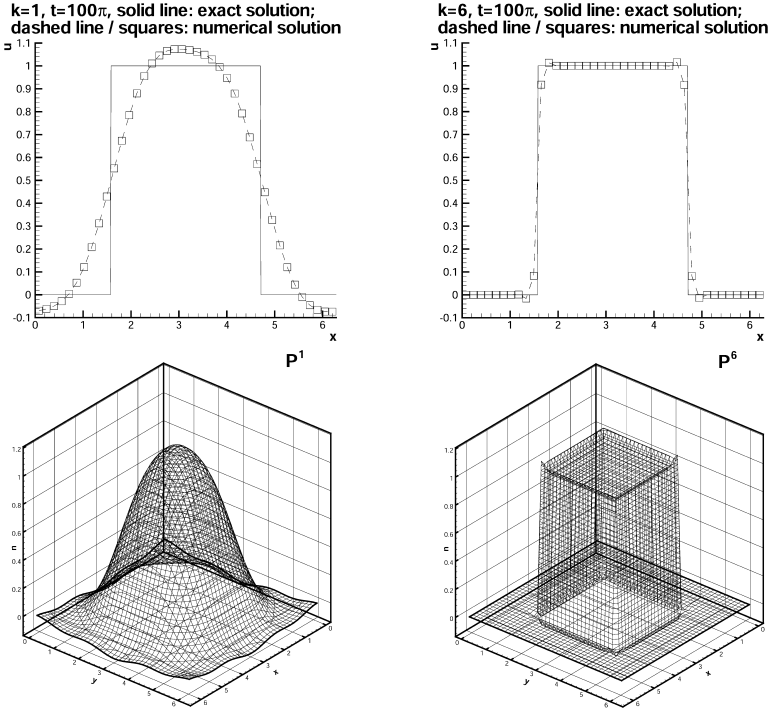

Fig. 1.2. Transport equation: Comparison of the exact and the RKDG solutions at $T = 100\pi$ with second order (P^1, left) and seventh order (P^6, right) RKDG methods. Top: one dimensional results with 40 cells, exact solution (solid line) and numerical solution (dashed line and symbols, one point per cell); Bottom: two dimensional results with 40×40 cells.

on the domain $(0, 2\pi) \times (0, T)$ or $(0, 2\pi)^2 \times (0, T)$ with the characteristic function of the interval $(\frac{\pi}{2}, \frac{3\pi}{2})$ or the square $(\frac{\pi}{2}, \frac{3\pi}{2})^2$ as initial condition and periodic boundary conditions; this is the case in which the dissipation of the scheme is going to be most noticeable since, unlike the previous case, the characteristics do not carry information into the discontinuity but *parallel* to it. To further accentuate the effect of the dissipation of the scheme, we compute the solution after a long time, namely, at $T = 100\pi$ (50 time periods). The results for the second and seventh order accurate RKDG methods are shown in Fig. 1.2, where we can clearly see that, as the accuracy increases, the dissipation decreases dramatically. These results are obtained without limiters. For linear problems, even with discontinuous solutions, limiters are usually not necessary for numerical stability (the numerical results are oscillatory but oscillations are reduced and localized when the order of the scheme increases). The result stays the same if we apply the TVB limiters detailed in Section 2 with the constant M suitably chosen.

• **Approximation of complex solutions**. Let us show that the RKDG method can handle solutions with very complicated structure. Consider the classical double-Mach reflection problem for the Euler equations of gas dynamics. In Fig. 1.3, by Cockburn and Shu [51], details of the approximation of the density are shown. Note that the strong shocks are very well resolved by the RKDG solution using piecewise linear and piecewise quadratic polynomials defined on *squares*. Also, note that there is a remarkable improvement in the approximation of the density near the contacts when going from linear to quadratic polynomials.

• **Curved boundaries**. Bassi and Rebay [19] showed the importance of approximating as accurately as possible the boundaries of the physical domain and the easiness with which this is achieved by using the RKDG methods. Indeed, for the classical two-dimensional isentropic flow around a circle, they showed that approximating the circle by a polygonal results in non-physical entropy production at each of the kinks which is then carried downstream and accumulate into a non-physical wake which does not disappear by further refining the grid. However, by simply taking into account the exact shape of the boundary, a remarkably improved approximation is obtained; see [19].

• **Parallelizability**. Finally, let us address the parallelizability of the RKDG method. In Table 1.1 below, taken from Biswas *et al.* [26], we display the results obtained by these authors; we see the solution time and total execution time for the two-dimensional problem

$$u_t + u_x + u_y = 0$$

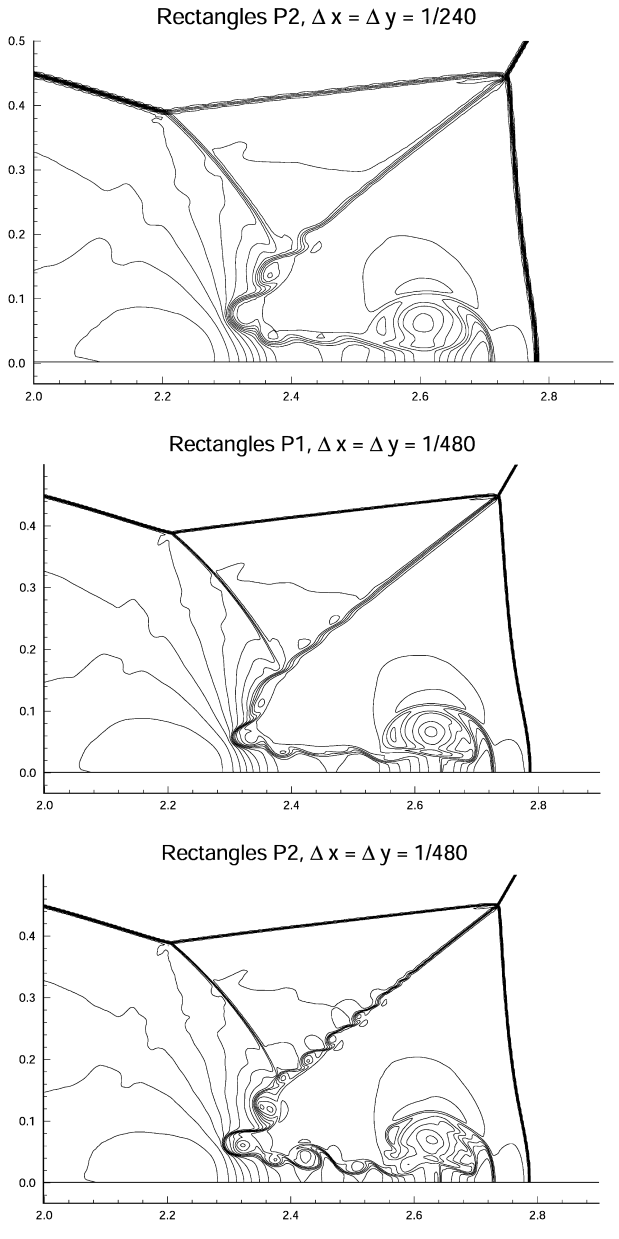

Fig. 1.3. Euler equations of gas dynamics: Double Mach reflection problem. Isolines of the density around the double Mach stems. Quadratic polynomials on squares $\Delta x = \Delta y = \frac{1}{240}$ (top); linear polynomials on squares $\Delta x = \Delta y = \frac{1}{480}$ (middle); and quadratic polynomials on squares $\Delta x = \Delta y = \frac{1}{480}$ (bottom).

Table 1.1. Scaled Parallel Efficiency. Solution Times (Without I/O) and Total Execution Times Measured on the nCUBE/2

Number of processors	Work (W)	Solution time (secs.)	Solution parallel efficiency	Total time (secs.)	total parallel efficiency
1	18,432	926.92	—	927.16	—
2	36,864	927.06	99.98%	927.31	99.98%
4	73,728	927.13	99.97%	927.45	99.96%
8	147,456	927.17	99.97%	927.58	99.95%
16	294,912	927.38	99.95%	928.13	99.89%
32	589,824	927.89	99.89%	929.90	99.70%
64	1,179,648	928.63	99.81%	931.28	99.55%
128	2,359,296	930.14	99.65%	937.67	98.88%
256	4,718,592	933.97	99.24%	950.25	97.57%

on the domain $(-\pi, \pi)^2 \times (0, T)$ with initial condition $u(x, y, 0) = \sin(\pi x)$ $\sin(\pi y)$ and periodic boundary conditions. Biswas *et al.* [26] used 256 elements per processor and ran the RKDG method with polynomials of degree two and 8 time steps; the work per processor was kept constant. Note how the solution time increases only slightly with the number of processors and the remarkable parallel efficiency of the method.

1.2. The LDG Space Discretizations for Convection-Diffusion Problems

The excellent results given by the RKDG methods for purely convective problems prompted several authors to try to extend them to the more complicated physical problems in which, although convection might be a dominating force, other physical phenomena must be taken into account. An early attempt, for example, was made in 1995 by Chen *et al.* [37] and by Chen *et al.* [36] in the framework of semiconductor device simulation; there, a DG space discretization was combined with *standard* mixed method elements for second-order elliptic problems.

In 1997, Bassi and Rebay [18] made a breakthrough in the framework of the compressible Navier–Stokes equations; they rewrote the equations as a first-order system and then discretized it by using the DG space discretization technique. Computations were performed for the laminar, subsonic flow around the NACA0012 airfoil at an angle of attack of zero degrees, free stream Mach number $M = 0.5$, and Reynolds number equal to 5000. The boundary is curved and the boundary layer is captured within only a few layers of elements; its separation at the trailing edge of the airfoil has been clearly resolved as can be seen in [18].

These remarkable results prompted the authors to introduce in 1998 the local discontinuous Galerkin (LDG) space discretization [50] by generalizing the method of Bassi and Rebay [18] and applying it to general convection-diffusion systems. The LDG method is in the same form as the general DG space discretization used for purely convective non-linear systems, with a different guiding principle for the choice of the numerical fluxes. When used with the special RK time discretizations and the generalized slope limiters described above, we obtain an RKDG method.

Of course, it is efficient to use RK time discretizations for convection-diffusion problems only if the convection is actually dominant, but, time discretizations and slope limiters aside, what the work of Bassi and Rebay [18] and Cockburn and Shu [50] showed is that DG discretizations could be used for a wide range of equations for which the DG methods had not been intended for originally, like, for example, purely elliptic equations.

1.3. Flexibility with the Mesh

Note also that, unlike any other finite element method, the RKDG methods can easily deal with meshes with hanging nodes and elements of several shapes since no inter-element continuity is required. This renders them particularly well suited for *hp*-adaptivity and for handling situations in which non-matching grids are necessary. These features have attracted the attention of many researchers who are currently vigorously studying and applying them (and other DG methods) to a wide variety of problems.

1.4. The Organization of this Review

In this paper, we expand the brief presentation of the discontinuous Galerkin method displayed in this section. We begin in Section 2, by describing in full detail the RKDG methods for non-linear scalar hyperbolic conservation laws in one space dimension; this contains most of the key ideas of the RKDG methods. Extensions to systems of non-linear hyperbolic conservation laws in several space dimensions are then discussed in Section 3. In Section 4, we consider the LDG space discretization for elliptic equations with emphasis on the discretization of the Laplacian and the Stokes operators and, in Section 5, we consider the RKDG methods for convection-diffusion problems. In Section 6, we extend the RKDG methods to Hamilton–Jacobi and non-linear second-order parabolic equations. We end in Section 7 by describing ongoing work and several important open problems.

Our purpose has been to describe and summarize in a single paper, all the work that the authors, their collaborators and other researchers have

carried out during more than a decade in order to develop and analyze the RKDG method. Thus, Section 2 is mainly related to the papers [49] and [48]; Section 3 to the papers [46], [41] and [51]; Section 4 to the papers [32], [42], and [43]; Section 5 to [107], [50], [33], and [88]; Section 6 to [72], [82] and [71]; finally, Section 7 to [47].

Part of the material in Sections 2, 3, 5 and 6, has also been considered in the short introductory monograph to DG methods [38]. However, in this paper, such material has been rewritten, updated and made more complete, clear and concise; also, new results have been added. This is especially true for Sections 2 and 5. For a description of the historical development of DG methods, the reader is referred to the paper [44], and for an overview of the state of the art of DG methods up to May 99, to [45].

2. SCALAR HYPERBOLIC CONSERVATION LAWS IN ONE SPACE DIMENSION

In this section, we introduce and study the RKDG method for non-linear hyperbolic conservation laws. Following the traditional path in this field, we begin by considering the simple model Cauchy problem for the scalar non-linear conservation law

$$u_t + f(u)_x = 0, \quad \text{in } (0, 1) \times (0, T), \qquad u(x, 0) = u_0(x), \quad \forall x \in (0, 1), \tag{2.1}$$

with periodic boundary conditions. All the main ideas of the devising of the RKDG method are discussed in this section.

As is well known, the main difficulty of a numerical solution to (2.1) is the appearance of shocks even if the initial condition $u_0(x)$ is smooth. A good scheme for (2.1) would hopefully have the following properties:

- It is locally conservative.
- It is high-order accurate in smooth regions of the solution.
- It has sharp and monotone (non-oscillatory) shock transitions.
- The numerical solution should be self-similar, that is, it should remain invariant when both space x and time t are scaled by the same constant. Notice that this self-similarity is an important property held by the exact solution of (2.1) and should be maintained by the numerical solution whenever possible.

We start by considering the DG space discretization. Next, we introduce the special RK time discretization and show how its structure allows us to guarantee the stability of the whole method provided the stability of a generic intermediate step holds. Then, we study carefully the stability of the

intermediate step and construct a generalized slope limiter that enforces it without degrading the high-order accuracy of the method. We then put all these elements together, show that the RKDG method is indeed stable and display several numerical results illustrating key features of the method. We end this section by extending the method to the bounded domain case.

2.1. The DG Space Discretization

We seek an approximation u_h whose restriction to each element $I_j = (x_{j-1/2}, x_{j+1/2})$ is, for each value of the time variable, an element of the local space $\mathscr{U}(I_j)$; typically, $\mathscr{U}(I_j)$ is the space of polynomials of degree at most $k \geqslant 0$. A reasonable way to define the initial data $u_h(\cdot, 0)$ on the element I_j is to take the L^2-projection of u_0 on the local space $\mathscr{U}(I_j)$, that is, for all $v_h \in \mathscr{U}(I_j)$,

$$\int_{I_j} u_h(x, 0)\, v_h(x)\, dx = \int_{I_j} u_0(x)\, v_h(x)\, dx \tag{2.2}$$

To determine the approximate solution for $t > 0$, we enforce the non-linear conservation law element-by-element by means of a Galerkin method. Thus, on each interval $I_j = (x_{j-1/2}, x_{j+1/2})$, we require that, for all $v_h \in \mathscr{U}(I_j)$,

$$\int_{I_j} (u_h(x, t))_t\, v_h(x)\, dx - \int_{I_j} f(u_h(x, t))(v_h(x))_x\, dx + \hat{f}(u_h(\cdot, t))\, v_h \Big|_{x_{j-1/2}}^{x_{j+1/2}} = 0 \tag{2.3}$$

where $\hat{f}(u_h)$ is the numerical flux. Note that u_h is a well defined function since there are as many equations per element as unknowns. The integral

$$\int_{I_j} f(u_h(x, t))(v_h(x))_x\, dx$$

could either be computed exactly or approximately by using suitable numerical quadratures or other methods; we will come back to this point in Section 3 when we discuss the multi-dimensional case. Thus, to complete the DG space discretization, we only have to define the numerical flux.

There are two main ideas in this crucial step. The first is to make the numerical flux depend *only* on the two values of the approximate solution u_h at the discontinuities, that is,

$$\hat{f}(u_h)(x_{j+1/2}) = \hat{f}(u_h(x_{j+1/2}^-), u_h(x_{j+1/2}^+))$$

this is computationally very convenient since we have a single recipe for the mapping $(a, b) \mapsto \hat{f}(a, b)$ regardless of the form of the local spaces $\mathscr{U}(I_j)$. Of course, we must make sure that the numerical flux is consistent with the non-linearity f it approximates and so we require that $\hat{f}(a, a) = f(a)$, The second idea is to pick the numerical flux in such a way that when the approximate solution u_h is piecewise-constant, the DG space discretization gives rise to a monotone finite volume scheme. The motivation for this is that, although only first-order accurate, monotone schemes are known to be stable and convergent to the exact solution; see the 1976 papers by Harten *et al.* [69] and by Kuznetsov [79] and the 1980 work by Crandall and Majda [52]. This is achieved by simply requiring that $a \mapsto \hat{f}(a, \cdot)$ be non-decreasing and $b \mapsto \hat{f}(\cdot, b)$ be non-increasing. The main examples of numerical fluxes satisfying the above properties are the following:

(i) The Godunov flux:

$$\hat{f}^G(a, b) = \begin{cases} \min\limits_{a \leqslant u \leqslant b} f(u), & \text{if} \quad a \leqslant b \\ \max\limits_{b \leqslant u \leqslant a} f(u), & \text{otherwise} \end{cases}$$

(ii) The Engquist–Osher flux:

$$\hat{f}^{EO}(a, b) = \int_0^b \min(f'(s), 0)\, ds + \int_0^a \max(f'(s), 0)\, ds + f(0)$$

(iii) The Lax–Friedrichs flux:

$$\hat{f}^{LF}(a, b) = \tfrac{1}{2}[f(a) + f(b) - C(b - a)]$$
$$C = \max_{\inf u^0(x) \leqslant s \leqslant \sup u^0(x)} |f'(s)|$$

This completes the definition of the DG space approximation.

Several comments about this DG space discretization are in order:

• The class of monotone schemes is one of the great achievements of the development of numerical schemes for non-linear scalar hyperbolic conservation laws. The stability and convergence properties of these schemes are corner stones for the construction of high-resolution finite volume and finite difference schemes. The same thing can be said about DG space discretizations which, as we have seen, try to capture those properties by incorporating their numerical fluxes.

• In the linear case $f(u) = cu$, all the above numerical fluxes coincide with the so-called *upwind* numerical flux, namely,

$$\hat{f}(a, b) = \begin{cases} ca & \text{if } c \geqslant 0 \\ cb & \text{if } c < 0 \end{cases}$$

The first DG method was introduced in 1973 by Reed and Hill [98] for a linear equation modeling transport of neutrons, namely,

$$\sigma u + \nabla \cdot (\mathbf{a}u) = f, \qquad \text{in } \Omega$$

They used the upwind numerical flux, even though the concept of monotone schemes and numerical fluxes had not been introduced yet.

• The above linear problem has been studied by several authors. The first error analysis of the DG method for this problem was carried out in 1974 by LeSaint and Raviart [83]. They proved that the $L^2(\Omega)$-norm of the error is of order k when polynomials of degree k are used for general triangulations; they also proved that, for tensor products of polynomials of degree k in one variable, the method super-converges with order $k+1$. Later in 1986, Johnson and Pitkaränta [77] showed that the method convergences with order $k+1/2$ for general triangulations and polynomials of degree k. In 1991, Peterson [97] numerically confirmed this rate to be optimal. In 1988, Richter [99] obtained the optimal order of convergence of $k+1$ for some *structured* two-dimensional non-Cartesian grids and polynomials of degree k. In 1996, Lin *et al.* [86] showed first order convergence for the DG method using piecewise-constant approximations. Their result holds for almost uniform grids of rectangles and for almost uniform grids of triangles. Their technique is based on a very interesting key approximation result; see also the review paper by Lin [85]. In 1993, Lin and Zhou [87] proved convergence to the weak solution assuming only that the exact solution belongs to $H^{1/2}(\Omega)$. More recently, Houston *et al.* [70] proved spectral convergence of the DG method assuming that the exact solution is piecewise analytic. A review of several techniques of analysis for finite element methods for hyperbolic problems including the DG method and the continuous Galerkin method can be found in the paper by Falk [59].

• The non-linear case is much more difficult to study. In fact, it was only in 1982, after the introduction of the monotone schemes, that Chavent and Salzano [35] used for the first time a DG space discretization for a non-linear hyperbolic conservation law in the framework of oil recovery problems. Moreover, so far there is no convergence analysis of the DG space discretization for non-linear scalar conservation laws with non-smooth

solutions except for the piecewise-constant case. Notice that the DG space discretization is linear (i.e. the scheme is linear for linear PDEs), hence it is expected to be oscillatory for problems with shocks except for the piecewise-constant case (the well known "Godunov Theorem", e.g. [84]). However, comparing with other high-order linear schemes such as finite difference and finite volume schemes, the method of lines version of the DG space discretization (and also certain implicit time discretization of it, such as backward Euler or Crank–Nicholson) satisfies a remarkably stronger provable stability property: a local cell entropy inequality for the square entropy as proven by Jiang and Shu in 1994 [74]. This result is valid for any order of accuracy and any triangulation in any spatial dimensions. It trivially implies a L^2 norm non-increasing with time for the numerical solution and enforces any limit solution to be the correct entropy solution when $f(u)$ is convex. The only comparable result for finite difference or finite volume schemes is for one space dimension, second order accurate only, *non-linear* schemes. However, unlike for the linear case, numerical evidence suggests that the DG method with explicit time stepping might be unstable in the general case with strong shocks.

• For non-linear problems, the best choice of numerical flux is the Godunov flux $\hat{f}^G$ since it is well-known that this is the numerical flux that produces the smallest amount of artificial viscosity. The local Lax–Friedrichs flux

$$\hat{f}^{LLF}(a, b) = \tfrac{1}{2}[f(a)+f(b)-C(b-a)], \qquad C = \max_{\min(a,b) \leqslant s \leqslant \max(a,b)} |f'(s)|$$

produces more artificial viscosity than the Godunov flux, but their performances are remarkably similar. Of course, if f is too complicated, we can always use the simple Lax–Friedrichs flux. Numerical experience suggests that as the degree k of the approximate solution increases, the impact of the choice of the numerical flux on the quality of the approximations decreases.

• In the special but important case in which the local space $\mathscr{U}(I_j)$ is taken to be the space of polynomials of degree k, the system of ordinary differential equations takes a particularly simple form if we choose the Legendre polynomials P_ℓ as basis functions because we can exploit their L^2-orthogonality, namely,

$$\int_{-1}^{1} P_\ell(s)\, P_{\ell'}(s)\, ds = \left(\frac{2}{2\ell+1}\right)\delta_{\ell\ell'}$$

to obtain a *diagonal* mass matrix. Indeed, if, for $x \in I_j$, we express our approximate solution u_h as follows:

$$u_h(x,t) = \sum_{\ell=0}^{k} u_j^\ell \varphi_\ell^j(x), \qquad \varphi_\ell^j(x) = P_\ell(2(x-x_j)/\Delta_j), \qquad \Delta_j = x_{j+1/2} - x_{j-1/2}$$

the initial condition (2.2) becomes

$$u_j^\ell(0) = \frac{2\ell+1}{\Delta_j} \int_{I_j} u_0(x)\, \varphi_\ell^j(x)\, dx$$

for $\ell = 0,\ldots, k$, and the weak formulation (2.3) takes the following simple form:

$$\frac{d}{dt} u_j^\ell(t) + \frac{2\ell+1}{\Delta_j} \left(- \int_{I_j} f(u_h(x,t))(\varphi_\ell^j(x))_x \, dx + \hat{f}(u_h(\cdot,t))\, \varphi_\ell^j \Big|_{x_{j-1/2}}^{x_{j+1/2}} \right) = 0$$

for $\ell = 0,\ldots, k$; moreover, note that $\varphi_\ell^j(x_{j+1/2}) = P_\ell(1) = 1$ and that $\varphi_\ell^j(x_{j-1/2}) = P_\ell(-1) = (-1)^\ell$.

• In the general case, the local mass matrix can be inverted easily, by means of a symbolic manipulator, for example, since its order is equal to the dimension of the local spaces. We thus can *always* obtain a system of ordinary differential equations for the degrees of freedom that we can write as follows:

$$\frac{d}{dt} u_h = L_h(u_h), \quad \text{in } (0,T), \qquad u_h(t=0) = \mathbb{P}_h u_0$$

where $\mathbb{P}_h$ denotes the L^2-projection and the function $L_h(u_h)$ is, of course, the approximation to $-f(u)_x$ provided by the DG-space discretization.

2.2. The RK Time Discretization

We discretize in time our system of ordinary differential equations by using the following RK method:

1. Set $u_h^{(0)} = u_h^n$;
2. For $i = 1,\ldots, \mathscr{K}$ compute the intermediate functions:

$$u_h^{(i)} = \sum_{l=0}^{i-1} \alpha_{il} w_h^{il}, \qquad w_h^{il} = u_h^{(l)} + \frac{\beta_{il}}{\alpha_{il}} \Delta t^n L_h(u_h^{(l)})$$

3. Set $u_h^{n+1} = u_h^{\mathscr{K}}$,

which is required to satisfy the following conditions:

(i) If $\beta_{il} \neq 0$ then $\alpha_{il} \neq 0$,

(ii) $\alpha_{il} \geqslant 0$,

(iii) $\sum_{l=0}^{i-1} \alpha_{il} = 1$.

We need to stress the following features of this special class of RK methods:

• Note that these RK methods are extremely simple to code since only a single routine for $L_h(u_h)$ needs to be written. Moreover, the evaluation of $L_h(u_h)$ can be efficiently done in parallel not only because the mass matrix can always be taken to be the identity but because when computing the restriction of $L_h(u_h)$ to the element K, only information of u_h of the neighbors sharing edges (in 2D) or faces (in 3D) with K is needed. This remains true regardless of the degree of the polynomial approximation and the accuracy in time of the RK method.

• Note that the first property allows us to express the RK method in terms of the functions w_h^{il}. Together with the two other properties this ensures the distinctive feature of these RK methods which is that their stability follows from the stability of the mapping $u_h^{(l)} \mapsto w_h^{il}$. Indeed, if we assume that, for some arbitrary semi-norm $|\cdot|$, we have that $|w_h^{il}| \leqslant |u_h^{(l)}|$, then

$$
\begin{aligned}
|u_h^{(i)}| = & \left| \sum_{l=0}^{i-1} \alpha_{il} w_h^{il} \right| \\
\leqslant & \sum_{l=0}^{i-1} \alpha_{il} |w_h^{il}|, && \text{by the positivity property (ii)} \\
\leqslant & \sum_{l=0}^{i-1} \alpha_{il} |u_h^{(l)}|, && \text{by the stability assumption} \\
\leqslant & \max_{0 \leqslant l \leqslant i-1} |u_h^{(l)}|, && \text{by the consistency property (iii)}
\end{aligned}
$$

It is clear now that that the inequality $|u_h^n| \leqslant |\mathbb{P}_h u_0|$, $\forall n \geqslant 0$, follows from the above inequality by a simple induction argument.

• This elegant and simple class of RK methods was identified in 1988 by Shu [106] in the framework of finite difference methods for non-linear hyperbolic conservation laws and was called the Total Variation Diminishing (TVD) RK time discretizations because the total variation was used as the semi-norm $|\cdot|$. The TVD–RK methods were further developed by Shu and Osher in 1988 [108] and in 1989 [109] for the efficient implementation of the high-resolution essentially non-oscillatory (ENO) schemes

Table 2.1. TVD–RK Time Discretization Parameters

order	α_{il}	β_{il}	$\max\{\beta_{il}/\alpha_{il}\}$
2	1	1	1
	$\frac{1}{2}\ \frac{1}{2}$	$0\ \frac{1}{2}$	
3	1	1	
	$\frac{3}{4}\ \frac{1}{4}$	$0\ \frac{1}{4}$	1
	$\frac{1}{3}\ 0\ \frac{2}{3}$	$0\ 0\ \frac{2}{3}$	

for hyperbolic conservation laws. In 1998, Gottlieb and Shu [63] carried out an exhaustive study of these methods. In 2000, Gottlieb *et al.* [64] reviewed this class of time discretizations, with new results for linear problems, and renamed it as "strong stability preserving," which seems closer to the spirit of the method. Some of the RK methods in this class are displayed in the Table 2.1.

• It is essential to carry out a von Neumann stability analysis of the method for the linear case $f(u) = cu$ in order to know for what values of the number $\mathrm{CFL}_{\mathrm{L}^2}$, the condition

$$|c|\frac{\Delta t}{\Delta x} \leqslant \mathrm{CFL}_{\mathrm{L}^2}$$

ensures its L^2-stability. This condition has to be respected even for non-linear functions f since only under this condition are the round-off errors not amplified.

For example, for DG discretizations using polynomials of degree k and a $k+1$ stage RK method of order $k+1$ (which give rise to an $(k+1)$th order accurate method), we can take in practice

$$\mathrm{CFL}_{\mathrm{L}^2} = \frac{1}{2k+1}$$

Indeed, this can be trivially proven for $k = 0$ and was proven for $k = 1$ in [49]. Moreover, for $k \geqslant 2$, the number $\frac{1}{2k+1}$ is less than 5% smaller than numerically-obtained estimates of $\mathrm{CFL}_{\mathrm{L}^2}$. In Table 2.2, we display these $\mathrm{CFL}_{\mathrm{L}^2}$ numbers for a wide variety of time and space discretizations. The symbol "★" indicates that the method is *unstable* when the ratio $\Delta t/\Delta x$ is held constant. In such a case, the method is typically stable for Δt of the order of $(\Delta x)^{1+\varepsilon}$ for some $\varepsilon > 0$; for example $\varepsilon = 1/2$ for the forward Euler method and polynomials of degree one [34].

Table 2.2. The CFL_{L^2} Numbers for Polynomials of Degree k and RK Methods of Order v

k	0	1	2	3	4	5	6	7	8
$v=1$	1.000	★	★	★	★	★	★	★	★
$v=2$	1.000	0.333	★	★	★	★	★	★	★
$v=3$	1.256	0.409	0.209	0.130	0.089	0.066	0.051	0.040	0.033
$v=4$	1.392	0.464	0.235	0.145	0.100	0.073	0.056	0.045	0.037
$v=5$	1.608	0.534	0.271	0.167	0.115	0.085	0.065	0.052	0.042
$v=6$	1.776	0.592	0.300	0.185	0.127	0.093	0.072	0.057	0.047
$v=7$	1.977	0.659	0.333	0.206	0.142	0.104	0.080	0.064	0.052
$v=8$	2.156	0.718	0.364	0.225	0.154	0.114	0.087	0.070	0.057
$v=9$	2.350	0.783	0.396	0.245	0.168	0.124	0.095	0.076	0.062
$v=10$	2.534	0.844	0.428	0.264	0.182	0.134	0.103	0.082	0.067
$v=11$	2.725	0.908	0.460	0.284	0.195	0.144	0.111	0.088	0.072
$v=12$	2.911	0.970	0.491	0.303	0.209	0.153	0.118	0.094	0.077

• Finally, let us consider the issue of the stability of the intermediate mapping $u_h^{(l)} \mapsto w_h^{il}$, which is nothing but a simple Euler forward step applied to the DG space discretization. That such a step could be stable is certainly not evident. In fact, in 1989, Chavent and Cockburn [34] used a DG space discretization with piecewise-linear functions and discretized it in time by using the forward Euler scheme. For the linear case $f(u) = cu$, they proved that a standard von Neumann analysis shows that the method is *unconditionally unstable*. This implies that all the mappings $u_h^{(l)} \mapsto w_h^{il}$ are unstable in L^2 as soon as polynomials of degree bigger than or equal to one are used (see Cockburn and Shu [49]), even though the complete RKDG method might be L^2-stable.

The above arguments indicate that a weaker measure of stability has to be used for the mapping $u_h^{(l)} \mapsto w_h^{il}$ to be stable. The fact that monotone schemes are obtained when piecewise-constant approximations are taken suggests that semi-norms of the local means of the approximate solution could be a good candidate for achieving the sought stability. As we show next, this turns out to be the case.

2.3. The Stability of the Step $u_h \mapsto w_h = u_h + \delta L_h(u_h)$

Let us denote by $\bar{u}_j$ the mean of u_h on the interval I_j. If we set $v_h \equiv 1$ in Eq. (2.3), we obtain,

$$(\bar{u}_j)_t + (\hat{f}(u_{j+1/2}^-, u_{j+1/2}^+) - \hat{f}(u_{j-1/2}^-, u_{j-1/2}^+))/\Delta_j = 0$$

where $u^-_{j+1/2}$ denotes the limit from the left and $u^+_{j+1/2}$ the limit from the right. This shows that if we set w_h equal to the Euler forward step $u_h + \delta L_h(u_h)$, we obtain

$$(\bar{w}_j - \bar{u}_j)/\delta + (\hat{f}(u^-_{j+1/2}, u^+_{j+1/2}) - \hat{f}(u^-_{j-1/2}, u^+_{j-1/2}))/\Delta_j = 0 \tag{2.4}$$

When the approximate solution is piecewise-constant, we obtain a monotone scheme for small enough values of $|\delta|$ and, as a consequence, we have that the scheme is TVD, i.e.,

$$|\bar{w}_h|_{TV(0,1)} \leqslant |\bar{u}_h|_{TV(0,1)}$$

where

$$|\bar{u}_h|_{TV(0,1)} \equiv \sum_{1 \leqslant j \leqslant N} |\bar{u}_{j+1} - \bar{u}_j|$$

is the total variation of the local means. For general approximate solutions, we get an analogous result that tells us when the scheme is total variation diminishing in the means (TVDM) of the approximate solution by using the following Lemma due to Harten [68].

Proposition 2.1 (Harten's Lemma). If the scheme (2.4) can be written into the form

$$\bar{w}_j = \bar{u}_j + C_{j+1/2}(\bar{u}_{j+1} - \bar{u}_j) - D_{j-1/2}(\bar{u}_j - \bar{u}_{j-1}) \tag{2.5}$$

with $C_{j+1/2}$ and $D_{j-1/2}$ being arbitrary non-linear functions of $\bar{u}_j$, $\bar{u}_{j\pm1}$ and $u^\pm_{j\pm1/2}$ satisfying

$$C_{j+1/2} \geqslant 0, \qquad D_{j+1/2} \geqslant 0, \qquad C_{j+1/2} + D_{j+1/2} \leqslant 1 \tag{2.6}$$

then it is TVDM, namely

$$|\bar{w}_h|_{TV(0,1)} \leqslant |\bar{u}_h|_{TV(0,1)}$$

The proof follows easily when we take a forward difference on both sides of (2.5), sum over j and group terms on the right hand side taking into consideration (2.6).

In fact, it is easy to rewrite (2.4) in the form (2.5) with

$$C_{j+1/2} = -\delta \left(\frac{\hat{f}(u^-_{j+1/2}, u^+_{j+1/2}) - \hat{f}(u^-_{j+1/2}, u^+_{j-1/2})}{\bar{u}_{j+1} - \bar{u}_j} \right) \tag{2.7}$$

$$D_{j-1/2} = \delta \left(\frac{\hat{f}(u^-_{j+1/2}, u^+_{j-1/2}) - \hat{f}(u^-_{j-1/2}, u^+_{j-1/2})}{\bar{u}_j - \bar{u}_{j-1}} \right) \tag{2.8}$$

Thus, the coefficients $C_{j+1/2}$ and $D_{j-1/2}$ are non-negative if and only if the following sign conditions are satisfied

$$\text{sign}(u^+_{j+1/2} - u^+_{j-1/2}) = \text{sign}(\bar{u}_{j+1} - \bar{u}_j),$$
$$\text{sign}(u^-_{j+1/2} - u^-_{j-1/2}) = \text{sign}(\bar{u}_j - \bar{u}_{j-1})$$

by the monotonicity of the numerical flux $\hat{f}$. Once these two conditions are satisfied, the third condition in (2.6) becomes a simple restriction on the size of the parameter δ.

Since the DG space discretization method does not provide an approximate solution automatically satisfying the above sign conditions, it is necessary to *enforce* them by means of what will be called a generalized slope limiter, $\Lambda\Pi_h$.

2.4. The Generalized Slope Limiter

Next, we construct the operator $\Lambda\Pi_h$; set $u_h = \Lambda\Pi_h v_h$. We begin by noting that for piecewise linear approximate solutions, that is,

$$v_h|_{I_j} = \bar{v}_j + (x - x_j)\, v_{x,j}$$

van Leer [111, 112] introduced the following slope limiter in the construction of his MUSCL schemes:

$$u_h|_{I_j} = \bar{v}_j + (x - x_j)\, m\left(v_{x,j}, \frac{\bar{v}_{j+1} - \bar{v}_j}{\Delta_j}, \frac{\bar{v}_j - \bar{v}_{j-1}}{\Delta_j}\right)$$

where the *minmod* function m is defined as follows:

$$m(a_1, a_2, a_3) = \begin{cases} s \min\limits_{1 \leqslant n \leqslant 3} |a_n| & \text{if} \quad s = sign(a_1) = sign(a_2) = sign(a_3) \\ 0 & \text{otherwise} \end{cases} \tag{2.9}$$

We use a less restrictive slope limiter, denoted by $\Lambda\Pi^1_h$, due to Osher [93], which is defined as follows:

$$u_h|_{I_j} = \bar{v}_j + (x - x_j)\, m\left(v_{x,j}, \frac{\bar{v}_{j+1} - \bar{v}_j}{\Delta_j/2}, \frac{\bar{v}_j - \bar{v}_{j-1}}{\Delta_j/2}\right)$$

which can be rewritten as follows:

$$u^-_{j+1/2} = \bar{v}_j + m\,(\, v^-_{j+1/2} - \bar{v}_j,\ \bar{v}_j - \bar{v}_{j-1},\ \bar{v}_{j+1} - \bar{v}_j) \tag{2.10}$$
$$u^+_{j-1/2} = \bar{v}_j - m\,(\, \bar{v}_j - v^+_{j-1/2},\ \bar{v}_j - \bar{v}_{j-1},\ \bar{v}_{j+1} - \bar{v}_j) \tag{2.11}$$

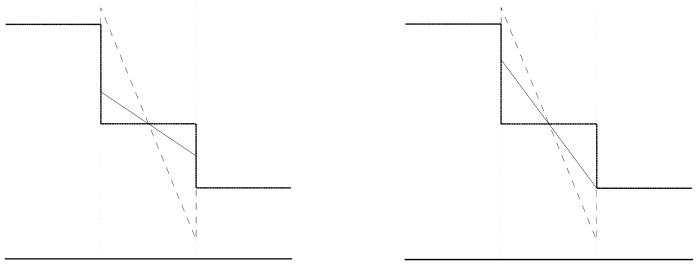

Fig. 2.1. Example of slope limiters: The MUSCL limiter (left) and the less restrictive $\Lambda\Pi_h^1$ limiter (right). Displayed are the local means of u_h (thick line), the linear function u_h in the element of the middle before limiting (dotted line) and the resulting function after limiting (solid line).

A comparison between the van Leer's MUSCL slope limiter and the slope limiter $\Lambda\Pi_h^1$ is displayed in Fig. 2.1.

For general functions v_h, we can define a generalized slope limiter $\Lambda\Pi_h$ in a very simple way. To do that, let us denote by v_h^1 the L^2-projection of v_h into the space of piecewise-linear functions. We then define $u_h = \Lambda\Pi_h(v_h)$ on the interval I_j, as follows:

(i) Compute $u_{j+1/2}^-$ and $u_{j-1/2}^+$ by using (2.10) and (2.11),

(ii) If $u_{j+1/2}^- = v_{j+1/2}^-$ and $u_{j-1/2}^+ = v_{j-1/2}^+$ set $u_h|_{I_j} = v_h|_{I_j}$,

(iii) If not, take $u_h|_{I_j}$ equal to $\Lambda\Pi_h^1(v_h^1)$.

Let us discuss some important points about this limiter:

• The above recipe is remarkably simple as it can be applied to any type of approximate solution v_h by using the minmod function m at most three times per element.

• This generalized slope limiter can be efficiently parallelized since to compute it on the element I_j, the only information needed which is not associated with this element are the means on the two neighboring elements.

• For this generalized slope limiter, the sign conditions are satisfied for small enough values of $|\delta|$. In fact, we have the following result.

Proposition 2.2 (The TVDM property). Suppose that for $j = 1,\ldots, N$

$$|\delta| \left(\frac{|\hat{f}(a,\cdot)|_{Lip}}{\Delta_{j+1}} + \frac{|\hat{f}(\cdot,b)|_{Lip}}{\Delta_j} \right) \leqslant 1/2 \tag{2.12}$$

Then, if $u_h = \Lambda\Pi_h v_h$,

$$|\bar{w}_h|_{TV(0,1)} \leqslant |\bar{u}_h|_{TV(0,1)}$$

Proof. Since in our case, the coefficients $C_{j+1/2}$ and $D_{j-1/2}$ in (2.5) are given by equations (2.7) and (2.8), we can clearly see condition (2.6) in Harten's Lemma is satisfied. This is a simple consequence of the monotonicity of the flux $\hat{f}$, the CFL condition (2.12), and the definition of the limited quantities (2.10), (2.11). □

• If the function v_h is linear, then $u_h = v_h$. However, if the function v_h is a parabola, we have that $u_h \neq v_h$ near the critical point of v_h. This is an indication that the high-order accuracy of the RKDG method is maintained away from critical points but might be lost near them; this is what actually takes place in practice. Fortunately, it is possible to slightly modify our generalized slope limiter in such a way that the degradation of the accuracy at local extrema is avoided. To achieve this, we follow Shu [105] and modify the definition of the generalized slope limiters by simply replacing the *minmod* function m by the corrected *minmod* function $\bar{m}_j$ defined as follows:

$$\bar{m}_j(a_1, a_2, a_3) = \begin{cases} a_1 & \text{if} \quad |a_1| \leqslant M\Delta_j^2 \\ m(a_1, a_2, a_3) & \text{otherwise} \end{cases}$$

where M is, of course, an upper bound of the absolute value of the second-order derivative of the solution at local extrema. In the case of the nonlinear conservation laws under consideration, it is easy to see that, if the initial data is piecewise C^2, we can take

$$M = C \sup\{|(u_0)_{xx}(y)|, y : (u_0)_x(y) = 0\}$$

where for a uniform mesh we could take $C = 2/3$; see [48]. Fortunately, in practice, the numerical results are not very sensitive to the choice of this constant which can be taken fairly big without degrading the quality of the results. For the above modified generalized slope limiter, which we denoted by $\Lambda\Pi_{h,M}$, the TVDM property of Proposition 2.2 does not hold anymore. Instead, the mapping $u_h \mapsto w_h$ has total variation that can increase an amount proportional to $\Delta x \equiv \max_j \Delta_j$ at each intermediate step.

Proposition 2.3 (The TVBM property). Suppose that for $j = 1,\ldots, N$

$$|\delta| \left(\frac{|\hat{f}(a,\cdot)|_{Lip}}{\Delta_{j+1}} + \frac{|\hat{f}(\cdot,b)|_{Lip}}{\Delta_j} \right) \leqslant 1/2$$

Then, if $u_h = \Lambda\Pi_{h,M} v_h$, then

$$|\bar{w}_h|_{TV(0,1)} \leqslant |\bar{u}_h|_{TV(0,1)} + CM\,\Delta x$$

Note that the condition on δ is *independent* of the form that the approximate solution has in space.

• Ideally, the parameter M should be estimated solely by using the approximate solution. However, it is difficult to achieve a recipe good for both smooth extrema and shocks. We should point out that this TVB modification renders the scheme non self similar, as the spatial mesh size Δ_j explicitly appears in the scheme and M certainly changes when x and t are both scaled by the same constant. However, in practical calculations one observes that this modification takes very little effect near the shock, and ensures the limiter is not enacted near smooth extrema. Thus the resulting scheme is basically the same as a self similar TVDM scheme except for the recovery of the full order of accuracy near smooth extrema. This issue brings us to the generalized slope limiter devised in 1994 by Biswas *et al.* [26] which does *not* require any auxiliary parameter to be guessed. Unlike the generalized slope limiter we have presented, there is no known stability property for it; however, it performs very well and can be used for adaptivity purposes.

• We have used the total variation of the local means to devise our generalized slope limiter but stability in the L^∞-norm of the means is also enforced by this limiter; see [48] for details.

2.5. The Non-Linear Stability of the RKDG Method

Let us recall the complete RKDG method:

1. Set $u_h^0 = \Lambda\Pi_{h,M} \mathbb{P}_h u_0$;
2. For $n = 0,\ldots, L-1$:
   (a) Set $u_h^{(0)} = u_h^n$;
   (b) For $i = 1,\ldots, \mathscr{K}$ compute the intermediate functions:

$$u_h^{(i)} = \Lambda\Pi_{h.M}\left(\sum_{l=0}^{i-1} \alpha_{il} w_h^{il}\right), \qquad w_h^{il} = u_h^{(l)} + \frac{\beta_{il}}{\alpha_{il}} \Delta t^n L_h(u_h^{(l)})$$

   (c) Set $u_h^{n+1} = u_h^{\mathscr{K}}$.

For this method, we have the following stability result.

Theorem 2.4 (TVDM-stability of the RKDG method). Let each time step Δt^n satisfy the following CFL condition:

$$\max_{il} \left|\frac{\beta_{il}}{\alpha_{il}}\right| \Delta t^n \left(\frac{|\hat{f}(a,\cdot)|_{Lip}}{\Delta_{j+1}} + \frac{|\hat{f}(\cdot,b)|_{Lip}}{\Delta_j} \right) \leqslant 1/2 \tag{2.13}$$

Then we have

$$|\bar{u}_h^n|_{TV(0,1)} \leqslant |u_0|_{TV(0,1)} + CMQ, \qquad \forall n = 0,\ldots, L$$

where $L\,\Delta x \leqslant Q$.

Proof. From Proposition 2.3 with $\delta = \frac{\beta_{il}}{\alpha_{il}}/\Delta t^n$ and the CFL condition (2.13), we have that

$$|\bar{w}_h^{il}|_{TV(0,1)} \leqslant |\bar{u}_h^{(l)}|_{TV(0,1)} + CM\,\Delta x$$

Now, we have

$$\begin{aligned}
|\bar{u}_h^{(i)}|_{TV(0,1)} &= \left| \sum_{l=0}^{i-1} \alpha_{il} \bar{w}_h^{il} \right|_{TV(0,1)} \\
&\leqslant \sum_{l=0}^{i-1} \alpha_{il}\, |\bar{w}_h^{il}|_{TV(0,1)}, \qquad \text{since} \quad \alpha_{il} \geqslant 0 \\
&\leqslant \sum_{l=0}^{i-1} \alpha_{il}\, (|\bar{u}_h^{(l)}|_{TV(0,1)} + CM\,\Delta x), \\
&\qquad \text{by the above stability property} \\
&\leqslant \max_{0 \leqslant l \leqslant i-1} |\bar{u}_h^{(l)}|_{TV(0,1)} + CM\,\Delta x, \qquad \text{since} \quad \sum_{l=0}^{i-1} \alpha_{il} = 1
\end{aligned}$$

and, by induction,

$$|u_h^n|_{TV(0,1)} \leqslant |\mathbb{P}_h u_0|_{TV(0,1)} + CML\,\Delta x \leqslant |u_0|_{TV(0,1)} + CMQ$$

since $\mathbb{P}_h$ is the L^2-projection. This completes the proof. □

The above stability result, and its remarkably simple proof, require several comments:

• Note that for the linear case $f(u) = cu$, the CFL condition (2.13) becomes

$$|c| \frac{\Delta t}{\Delta x} \leqslant \mathrm{CFL}_{\mathrm{TV}} \equiv \frac{1}{2 \max \frac{\beta_{il}}{\alpha_{il}}}$$

In Table 2.3, we display these CFL numbers for the RKDG obtained by using polynomials of degree k and the methods of order $k+1$ considered in the Table 2.1 and compare them with the CFL numbers needed for L^2-stability. We can see that the restriction of the time step imposed by the TVBM property is *much weaker* than that required to achieve L^2-stability. However, it is the condition for L^2 stability that needs to be respected; otherwise, the round-off errors would get amplified and the high-order accuracy of the method would degenerate even though the RKDG method remains TVBM-stable.

• In the proof of the above result, it can be clearly seen how the DG space discretization, the RK time discretization and the generalized slope limiter are inter-twined just in the right way to achieve non-linear stability. This is why we must emphasize that although the DG space discretization of this method is an essential distinctive feature, the other two ingredients are of no less relevance.

• Indeed, both the DG space discretization and the slope limiter were known to Chavent and Cockburn [34] who used piecewise-linear approximations and the forward Euler time marching scheme and obtained a *stable* first-order accurate in time method. It was the use of the special RK time discretization that really allowed the RKDG method to become a stable and high-order accurate in time method. We must say also that there are anecdotal reports of other time discretizations that seem to work just fine. However, the fact remains that only with this special class of TVD–RK methods can the non-linear stability of the method actually be *proven*.

• Let us also stress the fact that the generalized slope limiter is likewise an essential ingredient of the method without which its stability *cannot* be guaranteed. Although our numerical experience indicates that second-order

Table 2.3. CFL Numbers for RKDG Methods of Order $k+1$

k	0	1	2
$\mathrm{CFL}_{\mathrm{TV}}$	1	1/2	1/2
$\mathrm{CFL}_{\mathrm{L}^2}$	1	1/3	1/5

RKDG methods using piecewise linear approximations seem to remain stable, this is certainly *not* the case for higher order RKDG methods. However, if it is known before hand that the exact solution is smooth, the generalized slope limiters are not necessary. For a short essay in which the role of the generalized slope limiter is argued to be *indispensable* for transient non-linear problems, see the work by Cockburn [39]; in which it is shown that the limiter plays the role of the so-called shock-capturing terms used in DG and streamline-diffusion methods.

• It is interesting to note that totally independently of the just described development of the RKDG methods, other authors have studied methods using DG space discretizations and RK time discretizations. Indeed, in 1989, Allmaras [6] introduced a DG method for the transient and steady Euler equations of gas dynamics an earlier version of which appeared in the 1987 paper by Allmaras and Giles [7]. He used piecewise-linear functions in space and a three-stage second-order Runge–Kutta time stepping method. Later, in 1992, Halt [65] extended Allmaras' work to higher degree polynomials and to general unstructured grids in two- and three-space dimensions. His numerical test cases include steady state test problems like the Ringleb flow, 2D airfoils and the 3D Onera M6 wing; see also the 1991 and 1992 papers by Halt and Agarwall [66] and [67], respectively.

• It is not difficult to use Theorem 2.4 to conclude, by using a discrete version of the Ascoli–Arzelá theorem, that from the sequence $\{\bar{u}_h\}_{\Delta x > 0}$, it is possible to extract a subsequence strongly converging in $L^\infty(0, T; L^1(0, 1))$ to a limit $u^\star$. That this limit is a weak solution of the non-linear conservation law can be easily shown. However, while there is ample numerical evidence that suggests that $u^\star$ is actually the entropy solution, this fact remains a very challenging theoretical open problem. There are no other significant theoretical results about RKDG methods. Theoretical results concerning other DG methods for the non-linear scalar hyperbolic conservation laws are the 1995 convergence result of Jaffré *et al.* [73] for the so-called shock-capturing DG methods, and the 1996 a posteriori and a priori error estimates of Gremaud and Cockburn [40] for the same methods. These methods can be proven convergent, but they are non self-similar in an essential way, that is, mesh size Δx-dependent terms are responsible for the control of oscillations near shocks, hence their practical value is more limited.

2.6. Computational Results

In this subsection, we display the performance of the RKDG schemes in two simple but typical test problems; the $\Lambda\Pi_{h,M}$ generalized slope limiter is used.

The first test is the transport equation in $(0, 2\pi) \times (0, T)$ with periodic boundary conditions:

$$u_t + u_x = 0, \qquad u(x, 0) = \begin{cases} 1 & \text{for} \quad \pi/2 < x < 3\pi/2 \\ 0 & \text{otherwise} \end{cases}$$

Our purpose is to show that (i) when the TVB constant $M = 0$, the scheme becomes TVDM for all polynomial degree approximations, and that (ii) when M increases, the artificial diffusion induced by the limiter decreases as the polynomial degree increases. This can actually be seen in Fig. 2.2; see also Fig. 2.1 in Section 1. Note that for polynomials of degree 6, the contact discontinuity is always captured with less than five elements, for any value of M, even at $T = 100\pi$!

The second numerical experiment and its discussion is quoted almost verbatim from [38]. We consider the standard Burgers equation in $(0, 1) \times (0, T)$ with periodic boundary conditions:

$$u_t + (u^2/2)_x = 0, \qquad u(x, 0) = u_0(x) = \tfrac{1}{4} + \tfrac{1}{2} \sin(\pi(2x - 1))$$

Our purpose is to show that (i) when the constant M is properly chosen, the RKDG method using polynomials of degree k is order $k+1$ in the uniform norm away from the discontinuities, that (ii) the appearance of discontinuities does not destroy the high-order accuracy of the method away from them, that (iii) it is computationally more efficient to use high-degree polynomial approximations, and that (iv) shocks are captured in a few elements without production of spurious oscillations.

The exact solution is smooth at $T = 0.05$ and has a well developed shock at $T = 0.4$. In Tables 2.4 and 2.5, we show the effect of the parameter M on the quality of the approximation for $k = 1$ and $k = 2$, respectively. It can be seen that when the TVDM generalized slope limiter is used, i.e., when we take $M = 0$, there is degradation of the accuracy of the scheme, whereas when the TVBM generalized slope limiter is used with a properly chosen constant M, i.e., when $M = 20 \geqslant 2\pi^2$, the scheme is uniformly high-order accurate.

In Table 2.6, we display the history of convergence of the RKDG method with $M = 20$ away from the discontinuity. We see that, as claimed, the presence of the shock does not degrade the accuracy of the method away from it.

Next, we compare the efficiency of the RKDG schemes for $k = 1$ and $k = 2$ for the case $M = 20$ and $T = 0.05$; the efficiency of a method is the inverse of the product of the L^1-error times the computational cost (CPU).

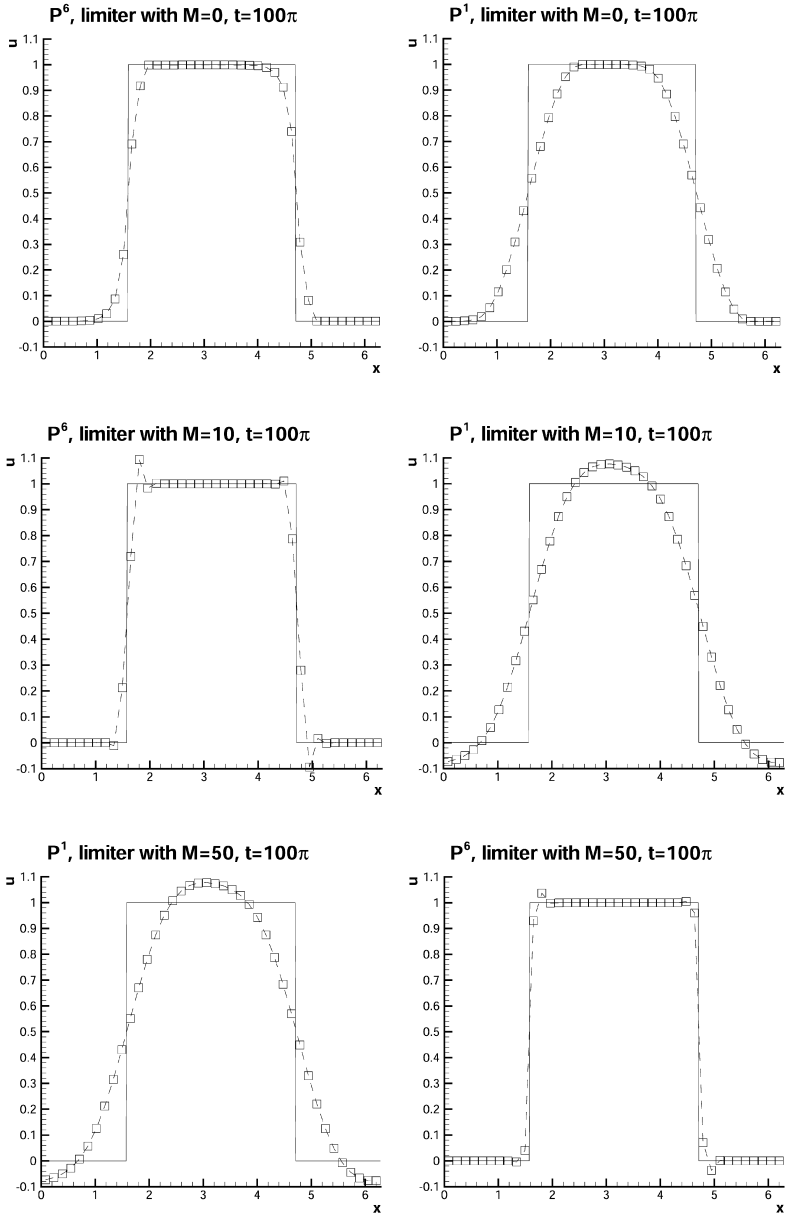

Fig. 2.2. Transport equation: Comparison of the exact and the RKDG solutions at $T = 100\pi$ with second order (P^1, left) and seventh order (P^6, right) RKDG methods with 40 elements. Exact solution (solid line) and numerical solution (dashed line and symbols, one point per element). The auxiliary constant M is 0 (top), 10 (middle), and 50 (bottom).

Table 2.4. Burgers Equation. Effect of the Generalized Slope Limiter Parameter M for $k = 1$ at $T = 0.05$

		$L^1(0, 1)$		$L^\infty(0, 1)$	
M	$1/\Delta x$	$10^5 \cdot$ error	order	$10^5 \cdot$ error	order
0	10	1286.23	—	3491.79	—
	20	334.93	1.85	1129.21	1.63
	40	85.32	1.97	449.29	1.33
	80	21.64	1.98	137.30	1.71
	160	5.49	1.98	45.10	1.61
	320	1.37	2.00	14.79	1.61
	640	0.34	2.01	4.85	1.60
	1280	0.08	2.02	1.60	1.61
20	10	1073.58	—	2406.38	—
	20	277.38	1.95	628.12	1.94
	40	71.92	1.95	161.65	1.96
	80	18.77	1.94	42.30	1.93
	160	4.79	1.97	10.71	1.98
	320	1.21	1.99	2.82	1.93
	640	0.30	2.00	0.78	1.86
	1280	0.08	2.00	0.21	1.90

The results, obtained on a Pentium II PC are displayed in Table 2.7. We can see that the efficiency of the RKDG scheme with quadratic polynomials is several times bigger than that of the RKDG scheme with linear polynomials even for very small values of Δx. We can also see that the efficiency ratio is proportional to $(\Delta x)^{-1}$, which is expected for smooth

Table 2.5. Burgers Equation. Effect of the Generalized Slope Limiter Parameter M for $k = 2$ at $T = 0.05$

		$L^1(0, 1)$		$L^\infty(0, 1)$	
M	$1/\Delta x$	$10^5 \cdot$ error	order	$10^5 \cdot$ error	order
0	10	2066.13	—	16910.05	—
	20	251.79	3.03	3014.64	2.49
	40	42.52	2.57	1032.53	1.55
	80	7.56	2.49	336.62	1.61
20	10	37.31	—	101.44	—
	20	4.58	3.02	13.50	2.91
	40	0.55	3.05	1.52	3.15
	80	0.07	3.08	0.19	3.01

Table 2.6. Burgers Equation. History of Convergence on $\Omega = \{x : |x - \text{shock}| \geqslant 0.1\}$ for $M = 20$ at $T = 0.4$

		$L^1(\Omega)$		$L^\infty(\Omega)$	
k	$1/\Delta x$	$10^5 \cdot$ error	order	$10^5 \cdot$ error	order
1	10	1477.16	—	17027.32	—
	20	155.67	3.25	1088.55	3.97
	40	38.35	2.02	247.35	2.14
	80	9.70	1.98	65.30	1.92
	160	2.44	1.99	17.35	1.91
	320	0.61	1.99	4.48	1.95
	640	0.15	2.00	1.14	1.98
	1280	0.04	2.00	0.29	1.99
2	10	786.36	—	16413.79	—
	20	5.52	7.16	86.01	7.58
	40	0.36	3.94	15.49	2.47
	80	0.06	2.48	0.54	4.84

solutions. This indicates that it is indeed more efficient to work with RKDG methods using polynomials of higher degree.

Finally, we have shown in Fig. 1.1 in Section 1 that when shocks are present, they can be captured in a few elements; in this case, the only shock is captured in essentially two elements, as is expected of any high-resolution method for strictly convex non-linearities. Note also that it is clear that the approximation using quadratic elements is superior to the approximation using linear elements. We also illustrate in Fig. 2.3 how the schemes follow a shock when it goes through a single element.

Table 2.7. Burgers Equation. Ratio of Efficiencies of the RKDG Method $(k = 2)/(k = 1)$ for $M = 20$ at $T = 0.05$

	$L^1(0, 1)$		$L^\infty(0, 1)$	
$1/\Delta x$	efficiency ratio	order	efficiency ratio	order
10	5.68	—	4.69	—
20	11.96	−1.07	31.02	−2.73
40	25.83	−1.11	70.90	−1.19
80	52.97	−1.04	148.42	−1.07

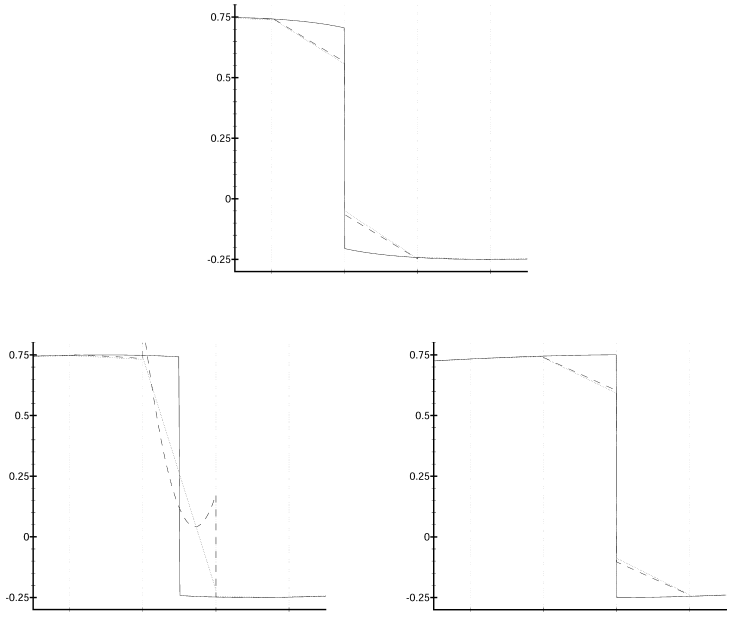

Fig. 2.3. Burgers equation. Comparison of the exact and the approximate solutions obtained with $M = 20$, $\Delta x = 1/40$ as the shock passes through one element. Exact solution (solid line), piecewise linear elements (dotted line) and piecewise quadratic elements (dashed line). Top: $T = 0.40$, lower left: $T = 0.45$, and lower right: $T = 0.50$.

2.7. Extension to Bounded Domains

In all this section, we have only considered periodic boundary conditions. To extend our results to the bounded domain case, see the formulation of the corresponding initial-boundary value problem by Bardos *et al.* [17], we proceed in three steps:

(i) To extend the DG space discretization, we simply have to replace the numerical fluxes at the boundaries, namely,

$$\hat{f}(u_h(0^-, t), u_h(0^+, t)) \qquad \text{and} \qquad \hat{f}(u_h(1^-, t), u_h(1^+, t))$$

by

$$\hat{f}(a(t), u_h(0^+, t)) \qquad \text{and} \qquad \hat{f}(u_h(1^-, t), b(t))$$

respectively, where $a(t)$ and $b(t)$ are the boundary data.

(ii) The TDV–RK time discretization has now to take into consideration the boundary data; see the work by Shu [104] for details.

(iii) To extend the generalized slope limiter, we simply have to define the quantities

$$\bar{u}_0 = 2a(t) - \bar{u}_1 \qquad \text{and} \qquad \bar{u}_{N+1} = 2b(t) - \bar{u}_N$$

and proceed as usual.

3. MULTI-DIMENSIONAL HYPERBOLIC SYSTEMS

In this section, we consider the extension of the RKDG method to multi-dimensional systems:

$$u_t + \nabla \cdot f(u) = 0, \qquad \text{in } \Omega \times (0, T) \tag{3.1}$$

$$u(x, 0) = u_0(x), \qquad \forall x \in \Omega \tag{3.2}$$

For simplicity, we assume that Ω is the d-dimensional unit cube. The RKDG method for multi-dimensional systems has the same structure it has for one-dimensional scalar conservation laws; we only need to describe the DG-space discretization and the generalized slope limiter $\Lambda\Pi_h$. After doing that, we display the performance of the method on the Euler equations of gas dynamics.

3.1. The Discontinuous Galerkin Space Discretization

To discretize the multi-dimensional system (3.1) in space, we simply proceed component by component; thus, it is enough to show how to do this for u being a scalar.

For this case, we seek an approximate solution u_h whose restriction to the element K of the triangulation $\mathcal{T}_h$ of Ω is, for each value of the time variable, in the local space $\mathcal{U}(K)$. Just as done in the one-dimensional case, we take $u_h(0) \equiv u_h(\cdot, 0)$ on the element K to be the L^2-projection of the data on $\mathcal{U}(K)$, i.e., for all $v_h \in \mathcal{U}(K)$,

$$\int_K u_h(0)\, v_h \, dx = \int_K u_0 v_h \, dx \tag{3.3}$$

We now determine the approximate solution for $t > 0$ on each element K of our triangulation by imposing that, for all $v_h \in \mathcal{U}(K)$,

$$\int_K (u_h)_t \, v_h \, dx - \int_K f(u_h) \cdot \nabla v_h \, dx + \int_{\partial K} \widehat{f \cdot n_K}(u_h) \, v_h \, ds = 0 \tag{3.4}$$

where n_K is the outward unit normal to the boundary of K.

To complete the definition of the DG space discretization, it only remains to define the numerical flux $\widehat{f \cdot n_K}$. This is in effect just a one dimensional flux we have discussed in the previous section, in the normal direction of the edge. However, to explain it clearly, we need to introduce some notation. For two adjacent elements K^+ and K^- of the triangulation $\mathcal{T}_h$ and a point x of their common boundary at which the vectors $n_{K^\pm}$ are well defined, we set

$$u_h^\pm(x) = \lim_{\varepsilon \downarrow 0} u_h(x - \varepsilon n_{K^\pm})$$

and call these values the traces of u_h from the *interior* of $K^\pm$. Now, just like for the one-dimensional case, we take the numerical flux at x to be solely a function of the traces $u_h^\pm(x)$, i.e.,

$$\widehat{f \cdot n_{K^-}}(u_h)(x) = \widehat{f \cdot n_{K^-}}(u_h^-(x), u_h^+(x))$$

and require that it be consistent with the non-linearity $f \cdot n_{K^-}$, which in this case amounts to ask that $\widehat{f \cdot n_{K^-}}(a, a) = f(a) \cdot n_{K^-}$. Another criterion to pick our numerical fluxes is that when a piecewise-constant approximation is taken, the DG space discretization should give rise to a monotone finite volume scheme. This is ensured if we ask that our numerical flux be *conservative*, i.e., that

$$\widehat{f \cdot n_{K^-}}(u_h^-(x), u_h^+(x)) + \widehat{f \cdot n_{K^+}}(u_h^+(x), u_h^-(x)) = 0$$

and that the mapping $a \mapsto \widehat{f \cdot n_{K^-}}(a, \cdot)$ be non-decreasing. The main examples of numerical fluxes satisfying all the above requirements are the following:

(i) The Godunov flux:

$$\widehat{f \cdot n}^G(a, b) = \begin{cases} \min\limits_{a \leqslant u \leqslant b} f \cdot n(u), & \text{if} \quad a \leqslant b, \\ \max\limits_{b \leqslant u \leqslant a} f \cdot n(u), & \text{otherwise} \end{cases}$$

(ii) The Engquist–Osher flux:

$$\widehat{f \cdot n}^{EO}(a, b) = \int_0^b \min(f' \cdot n(s), 0)\, ds + \int_0^a \max(f' \cdot n(s), 0)\, ds + f \cdot n(0)$$

(iii) The Lax–Friedrichs flux:

$$\widehat{f \cdot n}^{LF}(a, b) = \frac{1}{2}(f(a) + f(b)) \cdot n - \frac{C}{2}(b - a)$$

where $C = \max_{\inf u_0(x) \leqslant s \leqslant \sup u_0(x)} |n \cdot f'(s)|$.

In other words, to define the multi-dimensional DG discretization, we can use simple one-dimensional numerical fluxes.

Before discussing the DG discretization under consideration, we introduce a notation which is a mixture of the traditional notation used in hyperbolic conservation laws and that proposed in [29] for purely elliptic problems. Thus, we define the mean values $\{\cdot\}$ and jumps $[\cdot]$ by

$$\{u_h\} := \tfrac{1}{2}(u_h^+ + u_h^-), \qquad [u_h] := u_h^+ n_{K^-} + u_h^- n_{K^+}$$

we realize that we have the identity $\widehat{f \cdot n}^{LF} = \hat{f}^{LF} \cdot n$ where

$$\hat{f}^{LF}(u_h^-, u_h^+) = \{f(u_h)\} - \frac{C}{2}[u_h]$$

The Godunov and the Engquist–Osher numerical fluxes do not satisfy a similar identity.

Next, we discuss a few important points concerning this discretization:

• Just like in the one dimensional case, the mass matrix is block-diagonal; the block associated with the element K is a square matrix of order equal to the dimension of the local space $\mathscr{U}(K)$ and hence can be easily inverted. Moreover, for a variety of elements and spaces $\mathscr{U}(K)$, a basis can be found which is orthonormal in L^2. This is the case, for example, of rectangles and tensor product polynomials, in which case the orthonormal basis is a properly scaled tensor product of Legendre polynomials. Another remarkable example is that of simplexes and polynomials of a given total degree, case for which there is an orthonormal basis; see Dubiner [58], the work by Karniadakis and Sherwin [78] and Warburton [113], and the recent implementation by Aizinger *et al.* [5].

Thus, after performing the DG space discretization, and just like for the one-dimensional case, the resulting equations can be rewritten in ODE form as $\frac{d}{dt} u_h = L_h(u_h)$ where $L_h(u_h)$ denotes the approximation to $-\nabla \cdot f(u)$ provided by the DG method.

• In practice, the integrals appearing in the weak formulation (3.4) need to be approximated by quadrature rules. It was proven in [41] that

$$\|L_h(u) + \nabla \cdot f(u)\|_{L^\infty(K)} \leqslant C h^{k+1} |f(u)|_{W^{k+2,\infty}(K)}$$

if the quadrature rules over each of the faces of the border of the element K are exact for polynomials of degree $2k+1$, and if the one over the element is exact for polynomials of degree $2k$. In fact, these requirements are also necessary, as we have verified numerically; moreover, the method is more sensitive to the quality of the quadrature rules used on the boundary of the elements than to that used in their interior.

Finally, let us point out that a quadrature-free version of the method was devised by Atkins and Shu [12] which results in a very efficient method for linear problems and certain nonlinear problems such as Euler equations of gas dynamics where the nonlinearity in the flux is mainly low order polynomials and perhaps one or two divisions of the components of the independent variable u.

• When dealing with multi-dimensional hyperbolic systems, the local Lax–Friedrichs numerical flux is a particularly convenient choice of numerical flux because it can be easily applied to any non-linear hyperbolic system, it is simple to compute, and yields good results. This numerical flux is defined as follows. First, note that for multi-dimensional systems, u is a vector-valued function and $f(u)$ is a matrix whose rows will be denoted by $f_j(u)$; as a consequence, $\hat{f}^{LLF}$ is also a matrix whose jth. row is given by

$$\hat{f}_j^{LLF}(u_h^-, u_h^+) = \{f_j(u_h)\} - \frac{C}{2}[(u_h)_j]$$

where $C = C(K^\pm)$ is the larger one of the largest eigenvalue (in absolute value) of $\frac{\partial}{\partial u} f(u_h^\pm) \cdot n_{K^\pm}$. In practice, one could also determine $C = C(K^\pm)$ to be the larger one of the largest eigenvalue (in absolute value) of $\frac{\partial}{\partial u} f(\bar{u}_{K^\pm}) \cdot n_{K^\pm}$ where $\bar{u}_{K^\pm}$ are the means of the approximate solution u_h in the elements $K^\pm$.

• The DG space discretization can be applied to any high-order hyperbolic equation by simply rewriting it as a first order system of equations. For example, the wave equation

$$u_{tt} - c^2 \Delta u = 0$$

which is a second-order hyperbolic equation, can be rewritten as follows:

$$U_t + \nabla \cdot F(U) = 0$$

where,

$$U = \begin{pmatrix} q_1 \\ q_2 \\ \cdots \\ q_d \\ u \end{pmatrix}, \qquad F(U) = -\begin{pmatrix} u & 0 & \cdots & 0 \\ 0 & u & \cdots & 0 \\ \cdots & \cdots & \cdots & \cdots \\ 0 & 0 & \cdots & u \\ c^2 q_1 & c^2 q_2 & \cdots & c^2 q_d \end{pmatrix}$$

The DG space discretization can now be easily applied to this system.

- Let us finally point out that since the wave equation can be rewritten as

$$q_t - \nabla u = 0$$
$$u_t - c^2 \nabla \cdot q = 0$$

the DG space discretization of the hyperbolic system for U can also be rewritten in terms of (u, q) as follows: Find (u_h, q_h) such that its restriction to the element K belongs to the local space $\mathscr{U}(K) \times \mathscr{U}^d(K)$ and is such that, for all $(v_h, r_h) \in \mathscr{U}(K) \times \mathscr{U}^d(K)$,

$$\int_K (q_h)_t \cdot r_h \, dx + \int_K u_h \nabla \cdot r_h \, dx - \int_{\partial K} \hat{u}_h r_h \cdot n_K \, ds = 0$$

$$\int_K (u_h)_t \, v_h \, dx + \int_K c^2 q_h \cdot \nabla v_h \, dx - \int_{\partial K} \widehat{c^2 q_h} \cdot n_K v_h \, ds = 0$$

where the numerical fluxes $\hat{u}_h$ and $\widehat{c^2 q_h}$ can be easily written in terms of the numerical flux $\hat{F}(U_h)$. For example, the Lax–Friedrichs flux for $\hat{F}(U_h)$ corresponds to

$$-\hat{u}_h = -\{u_h\} - \frac{|c|}{2}[q_h], \qquad -\widehat{c^2 q_h} = -c^2\{q_h\} - \frac{|c|}{2}[u_h]$$

where

$$\{u_h\} = \tfrac{1}{2}(u_h^+ + u_h^-), \qquad [u_h] = u_h^+ n_{K^-} + u_h^- n_{K^+}$$
$$\{q_h\} = \tfrac{1}{2}(q_h^+ + q_h^-), \qquad [q_h] = q_h^+ \cdot n_{K^-} + q_h^- \cdot n_{K^+}$$

We have shown this to emphasize how easy is to discretize second-order elliptic operators by means of a DG method. We shall meet these numerical fluxes again when we deal with DG discretizations of purely elliptic equations in the next section.

3.2. The Generalized Slope Limiter $\Lambda\Pi_h$

When we dealt with the scalar one dimensional conservation law, the role of the generalized slope limiter $\Lambda\Pi_h$ was to enforce the TVBM property of a typical Euler forward time step. In the case of multi-dimensional scalar conservation laws, we cannot rely anymore on the TVBM property of the Euler forward step because such a property has not been proven for monotone schemes on general meshes; it has been proven only for monotone schemes in non-uniform but Cartesian grids in 1983 by Sanders [102]. We can, instead, rely on a local maximum principle. Indeed, in [41] Cockburn *et al.* constructed a generalized slope limiter that enforces a local maximum principle without degrading the accuracy of the numerical scheme; this property holds for approximate solutions of arbitrary shapes and quite general meshes. See also the limiters introduced and studied by Wierse [115].

After several years of numerical experimentation, the authors found a very simple, practical and effective generalized slope limiter $\Lambda\Pi_{h,M}$ which gives very good numerical results; see [51]. Since, unfortunately, there is no rigorous proof that the use of this limiter does enforce the stability of the method, we should at least provide the heuristics behind its construction. Let v_h be the function to which we are going to apply the limiter and let u_h be the result; let also v_h^1 be its L^2-projection into the space of piecewise linear functions. Inspired by the construction of the one-dimensional limiter described in Section 2, we first construct a slope limiter for piecewise linear functions, $\Lambda\Pi^1_{h,M}$. Then, we construct a limiter for general function as follows:

(i) Detect the spurious oscillations in $v_h|_K$,

(ii) If there is no spurious oscillation, set $u_h|_K = v_h|_K$,

(iii) If not, take $u_h|_K$ equal to $\Lambda\Pi^1_{h,M} v_h^1$.

It remains now to decide how to "detect the spurious oscillations." To do that, we *assume* that spurious oscillations are present in $v_h|_K$ on the element K only if they are present in $v_h^1|_K$ and by this we mean that $v_h^1|_K \neq \Lambda\Pi_{h,M} v_h^1|_K$. Thus, our generalized slope limiter is defined on the element K as follows:

(i) Compute $r_h|_K = \Lambda\Pi^1_{h,M} v_h^1|_K$,

(ii) If $r_h|_K = v_h^1|_K$, set $u_h|_K = v_h|_K$,

(iii) If not, set $u_h|_K = r_h|_K$.

It only remains to define the slope limiter $\Lambda\Pi^1_{h,M}$. To construct it for triangular elements, we proceed as follows; we quote almost verbatim the work done on pp. 133–134 in [38]. Consider the triangles in Fig. 3.1, where m_1 is the mid-point of the edge on the boundary of K_0 and b_i denotes the barycenter of the triangle K_i for $i = 0, 1, 2, 3$. Since

$$m_1 - b_0 = \alpha_1(b_1 - b_0) + \alpha_2(b_2 - b_0)$$

for some nonnegative coefficients α_1, α_2 which depend only on m_1 and the geometry, we can write, for any linear function v_h,

$$v_h(m_1) - v_h(b_0) = \alpha_1(v_h(b_1) - v_h(b_0)) + \alpha_2(v_h(b_2) - v_h(b_0))$$

and since

$$\bar{v}_{K_i} = \frac{1}{|K_i|}\int_{K_i} v_h = v_h(b_i), \qquad i = 0, 1, 2, 3$$

we have that

$$\tilde{v}_h(m_1, K_0) \equiv v_h(m_1) - \bar{v}_{K_0} = \alpha_1\,(\bar{v}_{K_1} - \bar{v}_{K_0}) + \alpha_2\,(\bar{v}_{K_2} - \bar{v}_{K_0}) \equiv \Delta\bar{v}(m_1, K_0)$$

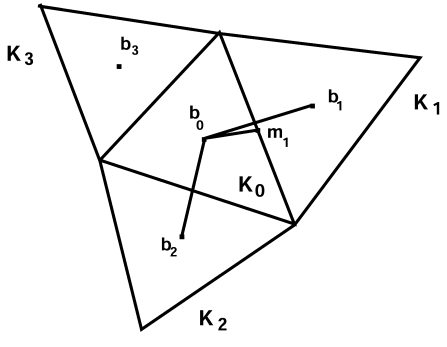

Fig. 3.1. Illustration of limiting.

Now, we are ready to describe the slope limiting. Let us consider a piecewise linear function v_h, and let m_i, $i = 1, 2, 3$, be the three mid-points of the edges of the triangle K_0. We can then write, for $(x, y) \in K_0$,

$$v_h(x, y) = \sum_{i=1}^{3} v_h(m_i)\, \varphi_i(x, y) = \bar{v}_{K_0} + \sum_{i=1}^{3} \tilde{v}_h(m_i, K_0)\, \varphi_i(x, y)$$

To compute $\Lambda\Pi_h^1 v_h$, we first compute the quantities

$$\Delta_i = \bar{m}(\tilde{v}_h(m_i, K_0), \nu\, \Delta\bar{v}(m_i, K_0))$$

where $\bar{m}$ is the TVB modified *minmod* function (without its third argument) and ν is an auxiliary parameter which we took equal to 2 in the one-dimensional case. Then, if $\sum_{i=1}^{3} \Delta_i = 0$, we simply set

$$\Lambda\Pi_{h,M}^1 v_h(x, y) = \bar{v}_{K_0} + \sum_{i=1}^{3} \Delta_i \varphi_i(x, y)$$

Note that if v_h is a linear function, then $\tilde{v}_h(m_i, K_0) = \Delta\bar{v}(m_i, K_0)$ and $\Delta_i = \tilde{v}_h(m_i, K_0)$ provided $\nu > 1$; in this case we have $\Lambda\Pi_h^1(v_h) = v_h$. This ensures that there is no degradation of accuracy after the application of the slope limiter away from critical points; when there are critical points, the suitable choice of the parameter M, hidden in the definition of the TVB modified *minmod* function, ensures the same effect.

If $\sum_{i=1}^{3} \Delta_i \neq 0$, we compute $pos = \sum_{i=1}^{3} \max(0, \Delta_i)$, $neg = \sum_{i=1}^{3} \max(0, -\Delta_i)$, and set $\theta^+ = \min\left(1, \frac{neg}{pos}\right)$, $\theta^- = \min\left(1, \frac{pos}{neg}\right)$. Then, we define

$$\Lambda\Pi_h v_h(x, y) = \bar{v}_{K_0} + \sum_{i=1}^{3} \hat{\Delta}_i\, \varphi_i(x, y)$$

where $\hat{\Delta}_i = \theta^+ \max(0, \Delta_i) - \theta^- \max(0, -\Delta_i)$.

For systems, limiting in the local characteristic variables gives remarkably superior results than doing it component-by-component. Thus, to limit the vector $\tilde{v}_h(m_i, K_0)$ in the element K_0, we proceed as follows:

- Find the matrix R and its inverse R^{-1}, which diagonalizes the Jacobian

$$J = \frac{\partial}{\partial u} f(\bar{v}_{K_0}) \cdot \frac{m_i - b_0}{|m_i - b_0|}$$

that is, $R^{-1} J R = \Lambda$, where Λ is a diagonal matrix containing the eigenvalues of J. Notice that the columns of R are the right eigenvectors of J and the rows of R^{-1} are the left eigenvectors.

- Transform $\tilde{v}_h(m_i, K_0)$ and $\Delta\bar{v}(m_i, K_0)$ to the characteristic fields. This is achieved by left multiplying these vectors by R^{-1}.
- Apply the scalar limiter to each of the components of the transformed vectors.
- The result is transformed back to the original space by multiplying R on the left.

3.3. Numerical Experiments

In what follows, we present some numerical results that display the performance of the method especially when applied to the Euler equations of gas dynamics. We show some numerical experiments with two objectives in mind. The first is to show that the use of polynomials of high degree is always beneficial. This is a well known fact that will be illustrated on the classical rotating hill test problem for scalar conservation laws. To show that this is also the case for solutions that display discontinuities, we consider the double-Mach reflection problem and show that the use of high degree polynomials not only does not degrade the approximation of strong shocks but furthermore provides a better approximation to contact discontinuities. The second objective is to show that to deal with singularities in the flow, we can use the typical finite element approach of adaptive refinement. To show this, we consider the forward facing step problem whose solution has a singularity right at the corner.

3.3.1. The Rotating Hill Problem

We display some of the numerical results reported in [5]. We consider the "rotating hill" problem

$$u_t - (2\pi y\, u)_x + (2\pi x\, u)_y = 0$$

with a "Gaussian hill" as initial data. We use polynomials of degree k on meshes of triangles. Given the mesh i, the mesh $i+1$ is obtained by dividing each triangle into four congruent triangles. In Fig. 3.2, we compare the linear solution on mesh 4, the quadratic solution on mesh 3, the cubic solution on mesh 2 and the quartic solution on mesh 1. All solutions are at $T = 1$, which represents one full rotation of the hill. We have taken the same temporal integration RK method and taken a small enough time step so that $(k+1)$th order of accuracy in the L^2-norm is achieved for $k = 1$ to $k = 6$.

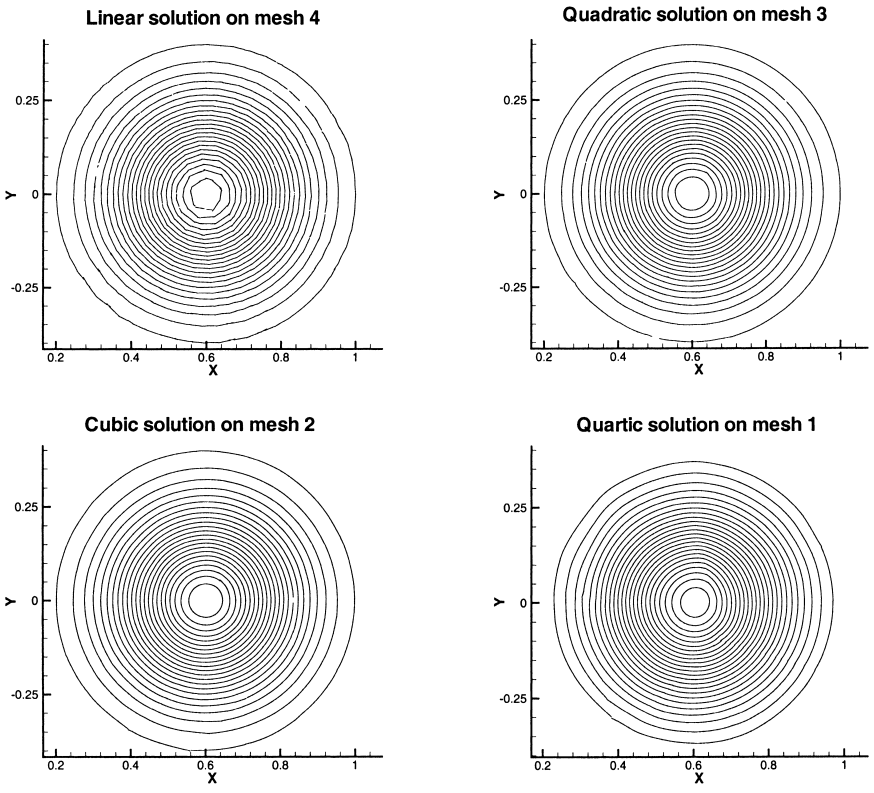

Fig. 3.2. Rotating hill problem. Comparison of different degree polynomials on different meshes.

In Fig. 3.3, we have plotted the L^2-error at time $T = 1$ versus the CPU time for the four different meshes described above and for polynomials of degree up to six. Each line corresponds to a different mesh, with the symbols on each line representing the error for the six different approximating spaces. We easily observe that exponential convergence is achieved and that it is always more efficient to use a coarser mesh with a higher order polynomial approximation.

3.3.2. The Double-Mach Reflection Problem

The results we show next are from Cockburn and Shu [51]. In Fig. 3.4, we display the contours of the density for different meshes for the third order scheme with P^2 elements. When comparing with the results obtained with P^1 elements (not shown here), we can observe a significant

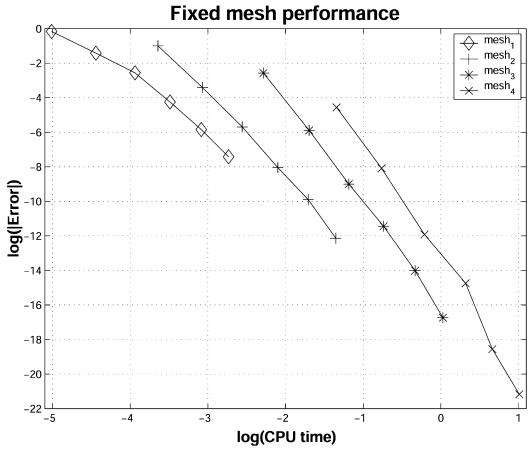

Fig. 3.3. Spectral convergence and comparison of L^2-error versus CPU time for 4 successively refined meshes and polynomials of degree 1 to 6.

improvement in the approximation of the density near the contacts. Next, we argue that the use of higher degree polynomials is more efficient. To better appreciate the difference between the P^1 and P^2 results, we show a "blowed up" portion around the double Mach region in Fig. 1.3 in Section 1. In Fig. 1.3, we can see that P^2 with $\Delta x = \Delta y = \frac{1}{240}$ has qualitatively the same resolution as P^1 with $\Delta x = \Delta y = \frac{1}{480}$, for the fine details of the complicated structure in this region. P^2 with $\Delta x = \Delta y = \frac{1}{480}$ gives a much better resolution for these structures than P^1 with the same number of rectangles. The conclusion here is that, if one is interested in the above mentioned fine structures, then one can use the third order scheme P^2 with only half the number of mesh points in each direction as in P^1. This translates into a reduction of a factor of 8 in space-time grid points for 2D time dependent problems, and will more than off-set the increase of cost per mesh point and the smaller CFL number by using the higher order P^2 method. This saving will be even more significant for 3D.

3.3.3. The Forward-Facing Step Problem

Again, the results we show next are from Cockburn and Shu [51]. The flow of a gas past a forward facing step is a problem studied extensively in Woodward and Colella [116] and later by many others. The main difficulty of this tests problem is the existence of a singularity in the solution located exactly at the corner of the step. It is well known that this leads to an erroneous entropy layer at the downstream bottom wall, as well as a spurious Mach stem at the bottom wall.

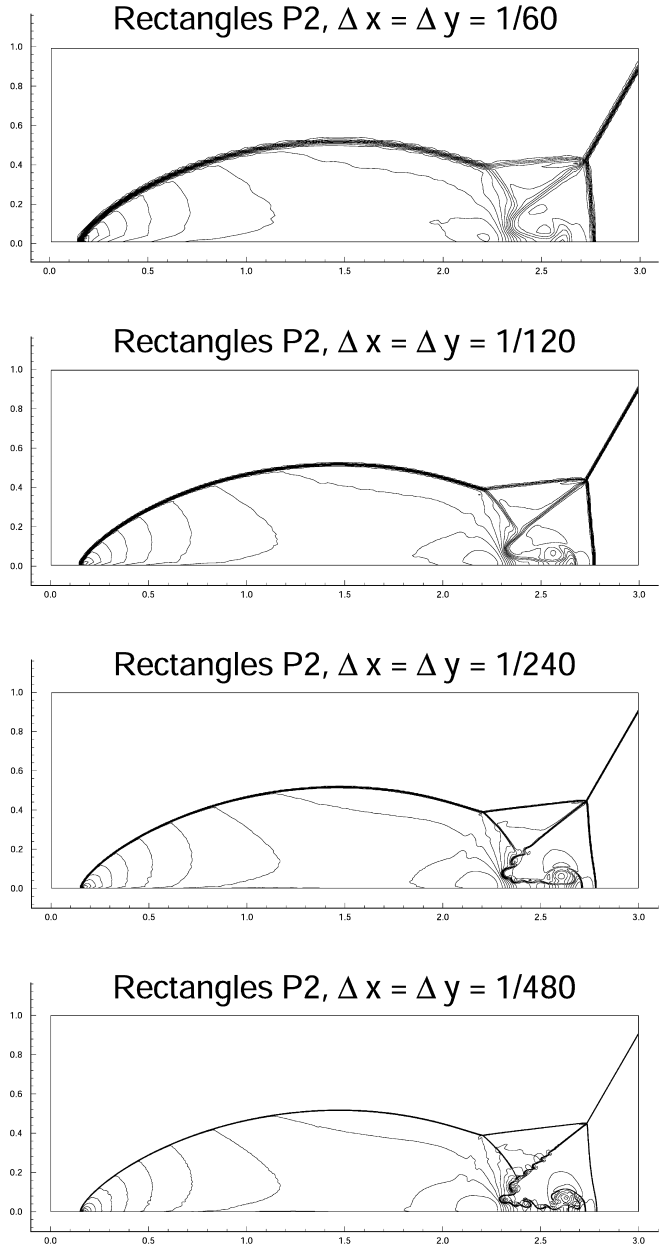

Fig. 3.4. Double Mach reflection problem. Third order P^2 results. Density ρ. 30 equally spaced contour lines from $\rho = 1.3965$ to $\rho = 22.682$. Mesh refinement study. From top to bottom: $\Delta x = \Delta y = \frac{1}{60}, \frac{1}{120}, \frac{1}{240}$, and $\frac{1}{480}$.

In Fig. 3.5, third order P^2 results using rectangle triangulation are shown, for a grid refinement study using $\Delta x = \Delta y = \frac{1}{40}$, $\Delta x = \Delta y = \frac{1}{80}$, $\Delta x = \Delta y = \frac{1}{160}$, and $\Delta x = \Delta y = \frac{1}{320}$ as mesh sizes. We can clearly see the improved resolution (especially at the upper slip line from the triple point) and decreased artifacts caused by the corner, with when the number of mesh points increases.

Next, we show that this singularity can be resolved by simply refining the grid around the corner and not by modifying our scheme near the corner in any way, as suggested in [116] and done in many other papers. We thus use our triangle code to locally refine near the corner progressively; we use the meshes displayed in Fig. 3.6. In Fig. 3.7, we plot the density obtained by the P^2 triangle code, with triangles (roughly the resolution of $\Delta x = \Delta y = \frac{1}{40}$, except around the corner). We can see that, with more triangles concentrated near the corner, the artifacts gradually decrease. Results with P^1 codes show a similar trend and hence is not shown here.

3.4. Concluding Remarks

In this section, we have extended the RKDG methods to multidimensional systems, and displayed the performance of the methods for the Euler equations of gas dynamics. The flexibility of the RKDG method to handle nontrivial geometries and to work with different elements has been displayed. Moreover, it has been shown that the use of polynomials of high degree not only does not degrade the resolution of strong shocks, but enhances the resolution of the contact discontinuities and renders the scheme more efficient on smooth regions.

Next, we extend the RKDG methods to convection-dominated problems. To do that, we start by considering the application of the DG space discretization to elliptic operators.

4. THE LDG DISCRETIZATION FOR ELLIPTIC PROBLEMS

In this section, we consider the LDG space discretization for second-order elliptic operators. This discretization technique is in the same spirit as that of the DG space discretization used for multi-dimensional hyperbolic systems which takes into account the elliptic nature of the operator in order to significantly reduce the computational complexity of the method; it achieves that by a suitable choice of the numerical fluxes.

We begin by considering the boundary value problem for the Laplace operator and by showing how to define the LDG discretization for this model elliptic problem. Then, we consider a boundary value problem for

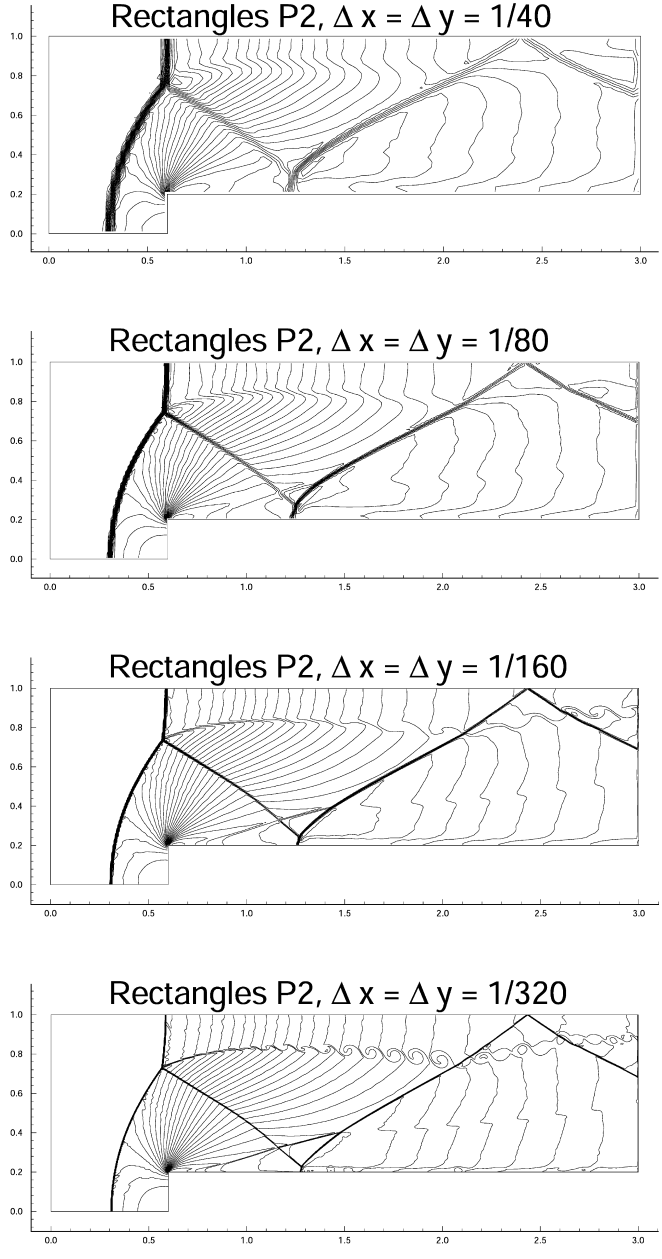

Fig. 3.5. Forward facing step problem. Third order P^2 results. Density ρ. 30 equally spaced contour lines from $\rho = 0.090338$ to $\rho = 6.2365$. Mesh refinement study. From top to bottom: $\Delta x = \Delta y = \frac{1}{40}, \frac{1}{80}, \frac{1}{160}$, and $\frac{1}{320}$.

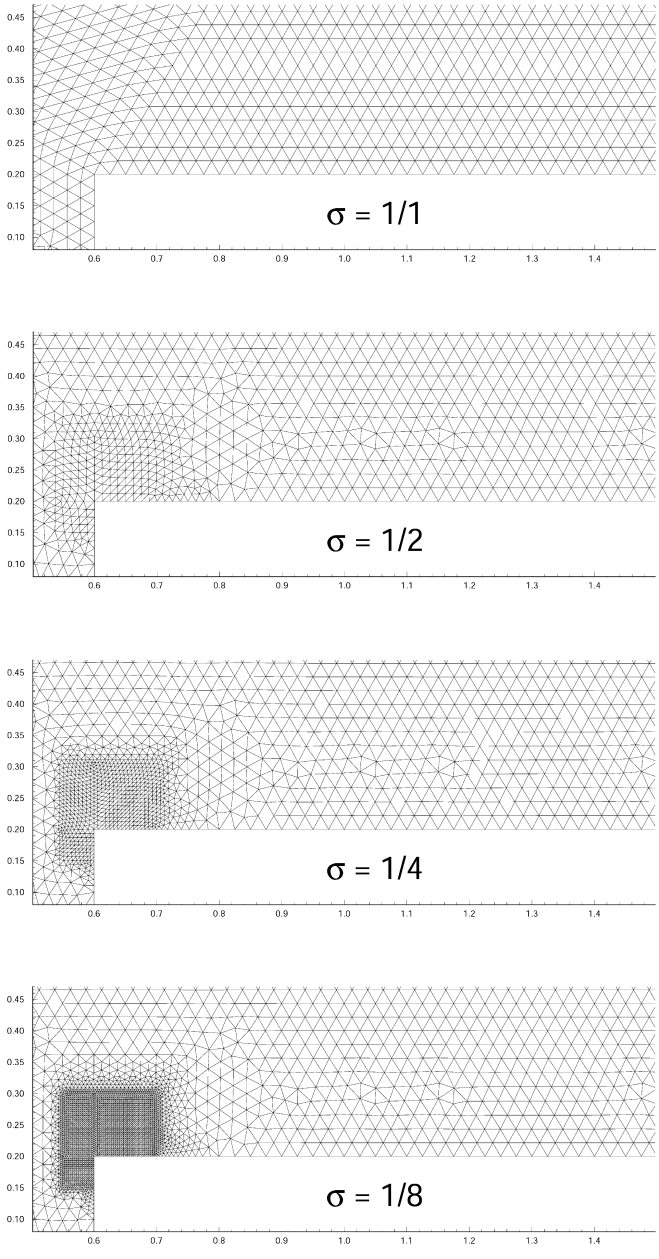

Fig. 3.6. Forward facing step problem. Detail of the triangulations associated with the different values of σ. The parameter σ is the ratio between the typical size of the triangles near the corner and that elsewhere.

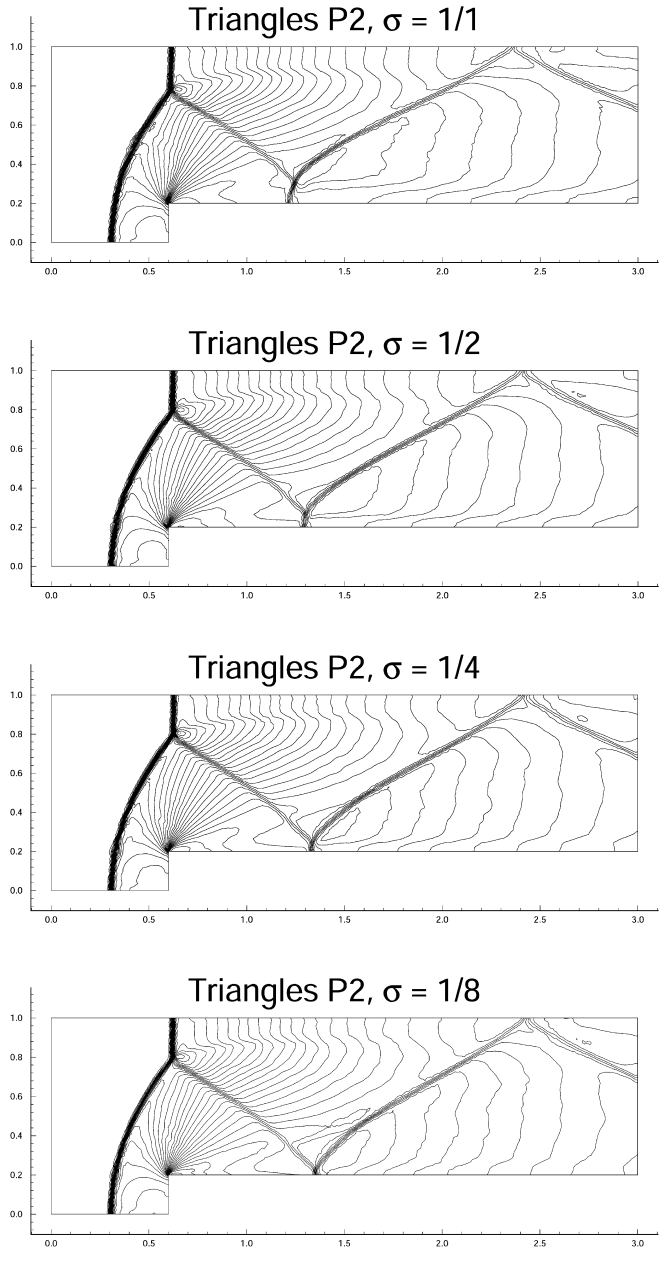

Fig. 3.7. Forward facing step problem. Third order P^2 results. Density ρ. 30 equally spaced contour lines from $\rho = 0.090338$ to $\rho = 6.2365$. Triangle code. Progressive refinement near the corner.

the Stokes system and show how to discretize it with an LDG method; here, our main purpose is to show how to deal with the incompressibility condition. We conclude the section by briefly comparing the LDG methods with stabilized mixed methods and with interior penalty methods.

4.1. The Laplacian

We begin by considering LDG methods for the classical model elliptic problem:

$$-\Delta u = f \quad \text{in } \Omega, \qquad u = g_{\mathscr{D}} \quad \text{on } \Gamma_{\mathscr{D}}, \qquad \frac{\partial u}{\partial n} = \mathbf{g}_{\mathscr{N}} \cdot \mathbf{n} \quad \text{on } \Gamma_{\mathscr{N}}$$

where Ω is a bounded domain of $\mathbb{R}^d$ and $\mathbf{n}$ is the outward unit normal to its boundary $\overline{\Gamma}_{\mathscr{D}} \cup \overline{\Gamma}_{\mathscr{N}}$.

4.1.1. The LDG Method

Just as we did for the wave equation, we rewrite our elliptic model problem as the following system of first-order equations:

$$\mathbf{q} = \nabla u, \quad -\nabla \cdot \mathbf{q} = f \quad \text{in } \Omega, \qquad u = g_{\mathscr{D}} \quad \text{on } \Gamma_{\mathscr{D}}, \qquad \mathbf{q} \cdot \mathbf{n} = \mathbf{g}_{\mathscr{N}} \cdot \mathbf{n} \quad \text{on } \Gamma_{\mathscr{N}}$$

Then, a general DG discretization is obtained as follows. The approximate solution $(\mathbf{q}_h, u_h)$ on the element K is taken in the space $\mathscr{Q}(K) \times \mathscr{U}(K)$ and is determined by imposing that for all $(\mathbf{r}, v) \in \mathscr{Q}(K) \times \mathscr{U}(K)$,

$$\int_K \mathbf{q}_h \cdot \mathbf{r} \, d\mathbf{x} = -\int_K u_h \nabla \cdot \mathbf{r} \, d\mathbf{x} + \int_{\partial K} \hat{u}_h \mathbf{r} \cdot \mathbf{n}_K \, ds,$$

$$\int_K \mathbf{q}_h \cdot \nabla v \, d\mathbf{x} = \int_K f v \, d\mathbf{x} + \int_{\partial K} v \hat{\mathbf{q}}_h \cdot \mathbf{n}_K \, ds$$

where $\hat{u}_h$ and $\hat{\mathbf{q}}_h$ are the *numerical fluxes* $\hat{u}_h$ and $\hat{\mathbf{q}}_h$. These are defined as follows. Inside the domain Ω, we take

$$\hat{\mathbf{q}}_h = \{\mathbf{q}_h\} + C_{11}[u_h] + \mathbf{C}_{12}[\mathbf{q}_h]$$
$$\hat{u}_h = \{u_h\} - \mathbf{C}_{12} \cdot [u_h] + C_{22}[\mathbf{q}_h]$$

and on its boundary, we take

$$\widehat{\mathbf{q}_h} := \begin{cases} \mathbf{q}_h^+ - C_{11}(u_h^+ - g_{\mathscr{D}})\,\mathbf{n} & \text{on } \Gamma_{\mathscr{D}} \\ \mathbf{g}_{\mathscr{N}} & \text{on } \Gamma_{\mathscr{N}} \end{cases}$$

$$\hat{u}_u := \begin{cases} g_{\mathscr{D}} & \text{on } \Gamma_{\mathscr{D}} \\ u_h^+ - C_{22}(\mathbf{q}_h^- - \mathbf{g}_{\mathscr{N}})\cdot\mathbf{n} & \text{on } \Gamma_{\mathscr{N}} \end{cases}$$

Several points have to be discussed about this method:

• Note how both the Dirichlet and Neumann boundary conditions are imposed through a suitable definition of the numerical fluxes.

• Note that if $C_{11} = 1/2$, $\mathbf{C}_{12} = 0$ and $C_{22} = 1/2$, we recover the Lax–Friedrichs numerical flux that we used to discretize in space the wave equation with the DG method. In the framework of the wave equation, the role of the parameters C_{11} and C_{22}, commonly thought of as inducing an *artificial viscosity*, is to render the method stable; in the elliptic case under consideration, they do have the same role. Moreover, for the method to be well defined, we must have that $C_{11} > 0$ and $C_{22} \geqslant 0$; the parameter $\mathbf{C}_{12}$ can be arbitrary.

• The LDG method is a particular case of the above general DG discretization technique for which the auxiliary parameter C_{22} is taken to be equal to zero. This reduces the stability of the LDG method but allows us to conveniently *eliminate* the auxiliary variable $\mathbf{q}$ from the equations in an element-by-element fashion; this local solvability is what gives the name to the LDG methods.

• The LDG method defines a unique solution under very mild compatibility condition on the local spaces $\mathscr{U}(K)$ and $\mathscr{Q}(K)$. In fact, it is enough to have that $\nabla\mathscr{U}(K) \subset \mathscr{Q}(K)$.

• When $\mathscr{U}(K)$ is the space of polynomials of degree $k \geqslant 1$ on each element and $\mathscr{Q}(K) = \mathscr{U}^d(K)$, Castillo *et al.* [32] proved that the rates of convergence of the L^2-norm of the error in u and $\mathbf{q}$ are of order $k+1$ and k, respectively, when the parameter C_{11} is taken to be of order h^{-1} and the parameters $\mathbf{C}_{12}$ are of order one. These orders of convergence were actually observed in the numerical experiments carried out in [32] on both structured and unstructured triangulations.

• When the parameter C_{11} is taken to be of order one only, it was proved [32] that order of convergence of u is $k+1/2$ and that of $\mathbf{q}$ is k. However, no degradation in the order of convergence from $k+1$ to $k+1/2$ was observed in the numerical experiments reported in [32]. Concerning this point, it is interesting to recall that the order of convergence of u for

the DG method for purely convective problems is $k+1/2$; this was proven in 1986 by Johnson and Pitkäranta [77] and was numerically confirmed in 1991 by Peterson [97]. Whether or not a similar phenomenon is actually taking place for the LDG method in this elliptic case remains to be investigated.

• In Cartesian grids, Cockburn *et al.* [42] proved that for a special choice of numerical fluxes (for which C_{11} is of order one and $|\mathbf{C}_{12} \cdot \mathbf{n}| = 1/2$), the orders of convergence are $k+1$ and $k+1/2$ for the L^2-norm of the error of u and $\mathbf{q}$.

4.1.2. Numerical Results for the LDG Method

Next, we quote a couple of numerical experiments from [42]. We solve the model problem in an L-shaped domain with Dirichlet boundary conditions in two cases.

In the first case, the exact solution is a function u that belongs to $H^s(\Omega)$ only for $s \leqslant 5$. We use five meshes obtained as follows. The 0th mesh is an unstructured mesh of 22 elements; then the jth mesh is obtained from the $(j-1)$th by refining each triangle into four congruent triangles. In the jth columns of Table 4.1, we display the orders of convergence for the L^2-errors in u and in $\mathbf{q}$ estimated by using the $(j-1)$th and the jth meshes; we can see that we obtain the orders of convergence of $\min\{5, k+1\}$ and $\min\{4, k\}$, respectively.

In the second case, we take the following exact solution $u(r, \theta) = r^\gamma \sin(\gamma\theta)$ where $\gamma = 2/3$, and solve for the corresponding Dirichlet problem. For conforming finite element methods, it has been shown that the orders of convergence in the H^1 and L^2 norms are $\frac{2}{3}-\varepsilon$ and $\frac{4}{3}-\varepsilon$ for all $\varepsilon > 0$, respectively. The numerical results for the LDG method on the sequence of unstructured meshes described in the previous experiment are reported in Table 4.2. They show that the orders of convergence are those of the conforming case.

Table 4.1. Orders of Convergence for an H^5-Solution on an L-Shaped Domain

k	L^2-error in the gradient $\mathbf{q}$				L^2-error in the potential u			
1	0.8494	0.8581	0.9148	0.9530	2.0435	1.9542	1.9552	1.9714
2	1.7966	1.8441	1.9136	1.9550	3.0471	2.9694	2.9740	2.9844
3	2.6595	2.8369	2.9260	2.9644	4.0360	3.9693	3.9831	3.9916
4	2.6559	3.7667	3.8908	3.9571	5.0226	4.8793	4.9274	4.9528
5	2.7630	3.7978	3.8723	3.8912	5.9726	4.8779	4.8875	4.8739
6	3.0742	3.9120	4.0307	4.1347	6.3544	4.9983	5.0609	5.0898

Table 4.2. Orders of Convergence for a Non-Smooth Solution on an L-shaped Domain

k	L^2-error in the gradient $\mathbf{q}$				L^2-error in the potential u			
1	0.7818	0.6298	0.6420	0.6513	1.6098	1.5694	1.5793	1.5760
2	0.7794	0.6662	0.6665	0.6666	1.5610	1.5383	1.5014	1.4639
3	0.7362	0.6665	0.6666	0.6666	1.5015	1.4810	1.4449	1.4137
4	0.7139	0.6666	0.6666	0.6667	1.4715	1.4543	1.4215	1.3950
5	0.7016	0.6666	0.6666	0.6667	1.4535	1.4383	1.4083	1.3849
6	0.6941	0.6666	0.6666	0.6667	1.4408	1.4277	1.3998	1.3786

4.2. The Stokes System

Next, we consider the Stokes system, i.e.,

$$-\Delta \mathbf{u} + \nabla p = \mathbf{f}, \qquad \nabla \cdot \mathbf{u} = 0 \quad \text{in } \Omega, \qquad \mathbf{u} = \mathbf{g}_{\mathscr{D}} \quad \text{on } \partial\Omega$$

where Ω is a bounded domain of $\mathbb{R}^d$ and the Dirichlet datum satisfies the usual compatibility condition $\int_\Omega \mathbf{g}_{\mathscr{D}} \cdot \mathbf{n}\, ds = 0$, where $\mathbf{n}$ is the outward unit normal to $\partial\Omega$.

4.2.1. The LDG Method

To defined an LDG method for the Stokes system, we begin by rewriting it as a first-order system,

$$\boldsymbol{\sigma}_i = \nabla u_i, \qquad -\nabla \cdot \boldsymbol{\sigma}_i + \partial_i p = f_i, \qquad 1 \leqslant i \leqslant d, \qquad \nabla \cdot \mathbf{u} = 0 \quad \text{in } \Omega$$
$$\mathbf{u} = \mathbf{g}_{\mathscr{D}} \quad \text{on } \partial\Omega$$

where u_i denotes the ith component of the velocity $\mathbf{u}$. Now, we discretize the above equations by using the DG technique. We take the approximate solution $(\boldsymbol{\sigma}_h, \mathbf{u}_h, p_h)$ on the element K in the space $\mathscr{S}(K)^d \times \mathscr{U}(K)^d \times \mathscr{P}(K)$ and we determine it by requesting that, for $1 \leqslant i \leqslant d$, for all $(\tau, v, w) \in \mathscr{S}(K) \times \mathscr{U}(K) \times \mathscr{P}(K)$,

$$\int_K \boldsymbol{\sigma}_{ih} \cdot \boldsymbol{\tau}\, d\mathbf{x} = -\int_K u_{ih} \nabla \cdot \boldsymbol{\tau}\, d\mathbf{x} + \int_{\partial K} \hat{u}_{\sigma,\, ih}\, \boldsymbol{\tau} \cdot \mathbf{n}_K\, ds$$

$$\int_K (\boldsymbol{\sigma}_{ih} \cdot \nabla v - p_h \partial_i v)\, d\mathbf{x} - \int_{\partial K} (\hat{\boldsymbol{\sigma}}_{hi} \cdot \mathbf{n}_K v - \hat{p}_h v \mathbf{n}_{Ki})\, ds$$

$$= \int_K f_i v\, d\mathbf{x} - \int_K \mathbf{u}_h \cdot \nabla q\, d\mathbf{x} + \int_{\partial K} \hat{\mathbf{u}}_{p,\, h} \cdot \mathbf{n}_K q\, ds = 0$$

where the numerical fluxes are, on the interior of the domain,

$$\hat{\boldsymbol{\sigma}}_{ih} = \{\boldsymbol{\sigma}_{ih}\} + C_{11}[u_{ih}] + \mathbf{C}_{12}[\boldsymbol{\sigma}_{ih}]$$
$$\hat{\mathbf{u}}_{\sigma,ih} = \{\mathbf{u}_{ih}\} - \mathbf{C}_{12} \cdot [\mathbf{u}_{ih}]$$

and, on the boundary,

$$\hat{\boldsymbol{\sigma}}_{ih} = \boldsymbol{\sigma}_{ih}^{+} - C_{11}(u_{ih}^{+} - g_{\mathscr{D},i})\,\mathbf{n}, \qquad \hat{\mathbf{u}}_{\sigma,h} = \mathbf{g}_{\mathscr{D}}$$

The numerical fluxes associated with the incompressibility constraint, $\hat{\mathbf{u}}_{p,h}$ and $\hat{p}_h$, are defined by using an analogous recipe. In the interior of Ω, we take

$$\hat{\mathbf{u}}_{p,h} = \{\mathbf{u}_h\} - D_{11}[p_h] - \mathbf{D}_{12}[u_h]$$
$$\hat{p}_h = \{p_h\} + \mathbf{D}_{12} \cdot [p_h]$$

and on the boundary, we take

$$\hat{\mathbf{u}}_{p,h} = \mathbf{g}_{\mathscr{D}}, \qquad \hat{p}_h = p_h^{+}$$

This completes the definition of the LDG method for the Stokes system. Note that:

• Cockburn *et al.* [43] proved that the order of convergence of k is obtained for the L^2-norm of the error in p and $\boldsymbol{\sigma}_i$, and $k+1$ for the L^2-norm of the velocity *provided* polynomials of degree k are used to approximate the pressure p, the stresses $\boldsymbol{\sigma}_i$, and the velocity $\mathbf{u}$. These orders of convergence were observed in numerical experiments.

• If polynomials of degree $k-1$ are used to approximate the pressure p and the stress tensor σ_i, it was proved [43] that the above mentioned orders of convergence remain *invariant*. However, this method is *less efficient* than the one obtained by using the same approximation spaces for all the variables.

4.2.2. Numerical Results

Next, we quote some of the numerical experiments in [43]. Consider the Stokes system with $\Omega = (-1, 1)^2$ and take the right-hand side f and the Dirichlet boundary condition $g_{\mathscr{D}}$ such that the exact solution is

$$u_1(x, y) = -e^x(y \cos y + \sin y), \qquad u_2(x, y) = e^x y \sin y$$
$$p(x, y) = 2e^x \sin y$$

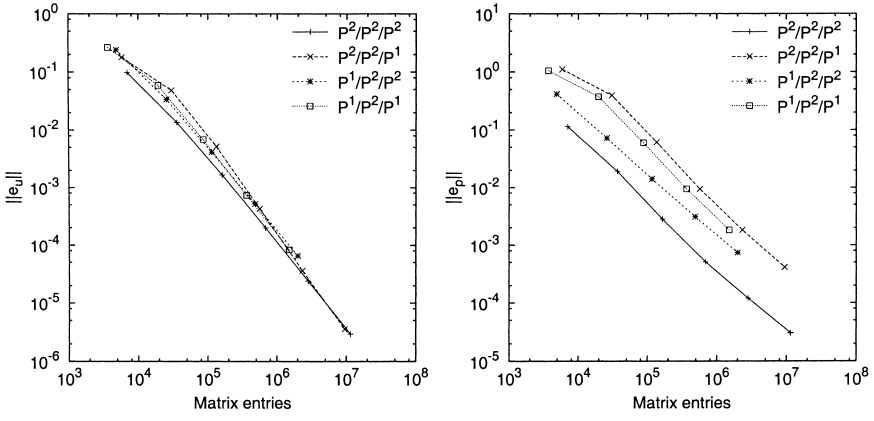

Fig. 4.1. Comparison of mixed spaces for quadratic (P^2) velocities.

We use uniform triangulations made of squares. The efficiency of LDG methods obtained with several combinations of local spaces is compared in Figs. 4.1 and 4.2. We can see that all these LDG discretizations converge with the same order, as expected, and that, in most cases, it is more efficient to use the same local approximating spaces for all quantities.

4.3. Relations with Other Methods

The LDG methods are closely related to interior penalty (IP) methods, to stabilized mixed methods and to mortar methods. Next, we briefly discuss the connection between these methods.

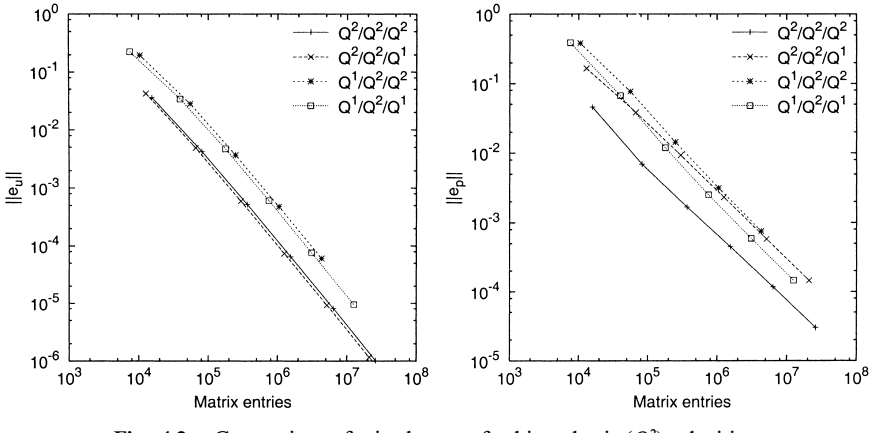

Fig. 4.2. Comparison of mixed spaces for biquadratic (Q^2) velocities.

• **Interior penalty methods**. Several IP methods were introduced and studied in the late 70's and early 80's. Thus, we have the IP method studied by Babu ška and Zlámal [14] in 1973 for fourth order problems, by Douglas and Dupont [57] in 1976 for second order elliptic and parabolic problems, by Baker [15] in 1977 also for fourth order problems, by Wheeler [114] in 1978 for second order elliptic problems, by Douglas *et al.* [56] in 1979 for non-linear hyperbolic equations, and by Arnold [9] in 1982 for linear and non-linear elliptic and parabolic problems. In [11], Arnold *et al.* showed that these IP methods and other DG methods for elliptic equations can be recast in a single framework. These other methods include the LDG method, the method of Baumann and Oden [22], the variations of the original method of Bassi and Rebay [18] studied by Brezzi *et al.* [29], and the variations of the method of Baumann and Oden introduced and studied by Rivière *et al.* [100].

For the model problem of the Laplacian, they showed that all these DG methods can be written as follows: $(\mathbf{r}, v) \in \mathcal{Q}(K) \times \mathcal{U}(K)$,

$$\int_K \mathbf{q}_h \cdot \mathbf{r}\, d\mathbf{x} = -\int_K u_h \nabla \cdot \mathbf{r}\, d\mathbf{x} + \int_{\partial K} \hat{u}_{h,K} \mathbf{r} \cdot \mathbf{n}_K\, ds$$

$$\int_K \mathbf{q}_h \cdot \nabla v\, d\mathbf{x} = \int_K f v\, d\mathbf{x} + \int_{\partial K} v \hat{\mathbf{q}}_{h,K} \cdot \mathbf{n}_K\, ds$$

which, of course, can be completely determined by their numerical fluxes $\hat{u}_{h,K}$ and $\hat{\mathbf{q}}_{h,K}$. Note that in this formulation, the numerical fluxes *can* have definitions that *might* depend on what side of the element boundaries we are; this is the case for the numerical fluxes in u of the last four schemes in Table 4.3 taken from [11]. In that table, the function $\alpha^r([u_h])$ is a special stabilization term introduced by Bassi and Rebay [21] and then identified

Table 4.3. Some DG Methods and Their Numerical Fluxes

Method	$\hat{\mathbf{q}}_{e,K}$	$\hat{u}_{h,K}$
Bassi–Rebay [18]	$\{\mathbf{q}_h\}$	$\{u_h\}$
Brezzi *et al.* [30]	$\{\mathbf{q}_h\} - \alpha^r([u_h])$	$\{u_h\}$
LDG [50]	$\{\mathbf{q}_h\} + C_{11}[u_h] - \mathbf{C}_{12}[\![\mathbf{q}_h]\!]$	$\{u_h\} + \mathbf{C}_{12} \cdot [u_h]$
IP [57]	$\{\nabla u_h\} + C_{11}[u_h]$	$\{u_h\}$
Bassi–Rebay [21]	$\{\nabla u_h\} - \alpha^r([u_h])$	$\{u_h\}$
Baumann–Oden [22]	$\{\nabla u_h\}$	$\{u_h\} - n_K \cdot [u_h]$
NIPG [100]	$\{\nabla u_h\} + C_{11}[u_h]$	$\{u_h\} - n_K \cdot [u_h]$
Babuška–Zlámal [14]	$C_{11}[u_h]$	$u_h\|_K$
Brezzi *et al.* [29]	$-\alpha^r([u_h])$	$u_h\|_K$

and studied by Brezzi *et al.* [29]. In [10], they refined their initial study and presented a much more complete study of these methods as well as a new unified error analysis.

Using an approach similar to the one introduced [11], a general theory of DG methods could be constructed for the Stokes. Let us just point out that here it is pertinent to distinguish between methods that impose the incompressibility condition weakly, like the LDG method we have presented here and methods that impose it pointwisely, like the 1990 method of Baker *et al.* [16] who use an IP discretization technique to achieve that goal.

• **Stabilized mixed methods**. Note that the above formulation for the Laplacian, based on the numerical fluxes, leads naturally to a mixed formulation. It is not very difficult to see that a suitable definition of the numerical fluxes induces a stabilization of the method which is associated with the *jumps* of the approximate solution across the element boundaries. For a discussion of the relation between LDG methods and stabilized mixed methods for the Stokes system, see [43].

• **Mortar methods**. The mortar methods, introduced in the papers by Bernardi *et al.* [25] and by Bernardi *et al.* [23, 24], were devised to allow the use of conforming methods in domains that were meshed independently of each other. The lack of continuity of the approximations across the boundaries of those domains is suitably controlled by the introduction of Lagrange multipliers which are now part of the unknowns. The DG methods can be considered to be mortar methods on each element for which the Lagrange multipliers, which are nothing but the numerical fluxes, are a given functions of the unknowns inside the elements. In [103], Schwab establishes an elegant link between mortar and DG methods.

4.4. Solvers

Solvers specifically designed for the linear system of equations given by DG methods have started to be developed. For the time-dependent compressible Navier–Stokes, Bassi and Rebay [20] experimented with the preconditioned GMRES and found that the simple block-Jacobi preconditioning was the most efficient. Recently, Feng and Karakashian [60] studied a domain decomposition preconditioner for DG approximations for purely elliptic problems. The condition number of their non-overlapping preconditioner grows linearly with the number of degrees of freedom in each subdomain. Later, Lasser and Toselli [81] found an overlapping domain decomposition method for DG methods for linear advection-diffusion problems whose condition number is independent of the number of degrees of freedom and

the number of subdomains. Another significant result has been recently obtained by Gopalakrishnan and Kanshat [62] who devised a multigrid method for solving the matrix equation of the IP method for elliptic problems. They proved that it convergences in a fixed number of iterations; they have also devised a method for the steady-state convection-diffusion problem which converges with a fixed number of iterations independently of the size of the convection coefficients.

4.5. Concluding Remarks

We have shown how to apply the LDG space discretization to second order elliptic model operators and how this is recast into the same form as the DG space discretization for multi-dimensional hyperbolic systems taking into consideration the elliptic nature of the operator for the choice of the numerical fluxes. We are now ready to continue our presentation of the RKDG method for convection-dominated problems. However, we want to stress that the application of LDG methods to linear elasticity, to the biharmonic equation, and to other elliptic problems constitute topics that are currently being vigorously studied.

5. CONVECTION-DIFFUSION EQUATIONS

In this section, we consider the solution of convection dominated convection diffusion equations using the RKDG method. The discussion will concentrate on the DG spatial discretization (method of lines) which we illustrate on a periodic boundary conditions setting; boundary conditions can be treated similar to the case of elliptic equations in the previous section.

5.1. A Simple Example and Basic Ideas

First, we would like to motivate the key ideas and to indicate a "pitfall" due to the presence of second order derivative diffusion terms, through the following initial value problem for the simple heat equation:

$$u_t - u_{xx} = 0 \quad \text{in } (0, 2\pi) \times (0, T), \qquad u(x, 0) = \sin(x) \quad \forall x \in (0, 2\pi)$$

with periodic boundary conditions; we follow Shu [107].

It seems that the most natural way of extending the DG spatial discretization (2.3) would be simply to replace the flux $f(u)$ by $-u_x$ and then proceed in a straightforward way. Thus, we take the restriction of

$u_h(\cdot, t)$ to each element I_j in the local space $\mathscr{U}(I_j)$, which we take to be polynomials of degree at most k, and define $u_h(\cdot, t)$ by asking that for all $v_h \in \mathscr{U}(I_j)$,

$$\int_{I_j} (u_h(x,t))_t \, v_h(x)\, dx + \int_{I_j} (u_h(x,t))_x \, (v_h(x))_x \, dx - \widehat{(u_h)_x}\,(\cdot, t)\, v_h \Big|_{x_{j-1/2}}^{x_{j+1/2}} = 0 \tag{5.1}$$

where, for the lack of up-winding mechanism in a heat equation one naturally takes a central numerical flux

$$\widehat{(u_h)_x}\,(x_{j+1/2}, t) = \tfrac{1}{2}\left((u_h)_x(x_{j+1/2}^-, t) + (u_h)_x\,(x_{j+1/2}^+, t)\right)$$

We remark that, in an actual computation, the scheme takes the simple form

$$\frac{d}{dt}(u_h)_j + \frac{1}{\Delta_j^2}\left(A(u_h)_{j-1} + B(u_h)_j + C(u_h)_{j+1}\right) = 0 \tag{5.2}$$

where $(u_h)_j$ is a small vector of length $k+1$ containing the coefficients of the solution u_h in the local basis inside the element I_j, and A, B, C are $(k+1)\times(k+1)$ constant matrices which can be computed once and for all and stored at the beginning of the code. Time discretization can be achieved by the same TVD Runge–Kutta methods discussed in Section 2.2.

We compute with the scheme (5.1) and show in Table 5.1 the L^2 and L^∞ errors and numerically observed orders of accuracy for the two cases $k=1$ and 2 (piecewise linear and piecewise quadratic cases) to $T=0.8$. Clearly there is an order one error for both cases which does not decrease with a mesh refinement! We plot the solutions with 160 cells in Fig. 5.1 and can clearly see that the computed solutions have completely incorrect amplitudes, i.e. the scheme is not consistent.

Table 5.1. L^2 and L^∞ Errors and Orders of Accuracy for the Inconsistent Discontinuous Galerkin Method (5.1) Applied to the Heat Equation $u_t = u_{xx}$ with Initial Condition $u(x, 0) = \sin(x)$, $T = 0.8$. Third Order Runge–Kutta in Time

	$k=1$				$k=2$			
h	L^2 error	order	L^∞ error	order	L^2 error	order	L^∞ error	order
$2\pi/20$	1.78E-01	—	2.58E-01	—	1.85E-01	—	2.72E-01	—
$2\pi/40$	1.76E-01	0.016	2.50E-01	0.025	1.78E-01	0.049	2.55E-01	0.089
$2\pi/80$	1.75E-01	0.004	2.48E-01	0.012	1.77E-01	0.013	2.51E-01	0.025
$2\pi/160$	1.75E-01	0.001	2.48E-01	0.003	1.76E-01	0.003	2.50E-01	0.007

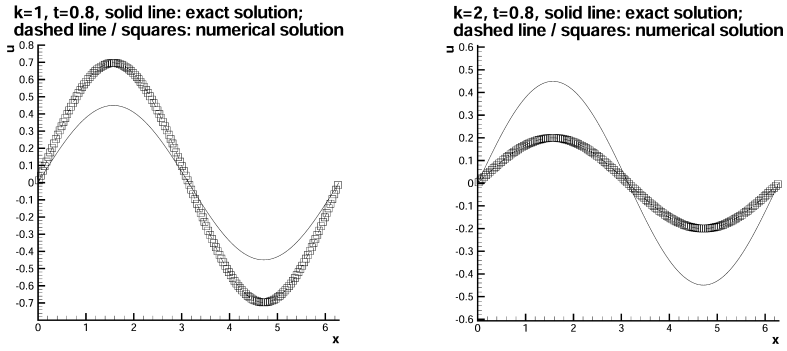

Fig. 5.1. The inconsistent discontinuous Galerkin method (6.1) applied to the heat equation $u_t = u_{xx}$ with an initial condition $u(x, 0) = \sin(x)$. $T = 0.8$. 160 cells. Third order Runge–Kutta in time. Solid line: the exact solution; dashed line and squares symbols: the computed solution at the cell centers. Left: $k = 1$; Right: $k = 2$.

This is a very subtle case of inconsistency: the exact solution of our model problem does satisfy the scheme (5.1) *exactly*. Hence one might base the judgment on one's experience with finite difference methods and conclude that the method is consistent. However, those familiar with non-conforming approximations of elliptic problems would remember that a similar type of inconsistency was present in one of the first papers on the subject, namely, the 1973 paper by Babuška and Zlámal [14]; such a "variational crime," see also Strang and Fix [110], could be controlled by the introduction of a term whose role was to "recover" the continuity of the approximation.

It is actually very dangerous that the scheme (5.1) produces a stable but completely incorrect solution. If one was in a hurry and did not want to do the ground work of either a theoretical proof of convergence or a testing of the method on the simple heat equation first which has a known exact solution, but rather went to solve the complicated Navier–Stokes equations and produced beautiful color pictures, one would not be able to tell that the result is actually wrong. In fact, the inconsistent scheme (5.1) has been used in the literature for discretizing the viscous terms in the Navier–Stokes equations.

On the other hand, if we rewrite the heat equation $u_t = u_{xx}$ as a first order system

$$u_t - q_x = 0, \qquad q - u_x = 0 \tag{5.3}$$

we can then *formally* use the same discontinuous Galerkin method as in Section 2 for the convection equation to solve (5.3), resulting in the following

scheme: find u_h, q_h, whose restriction to each element I_j are, for each t, elements of the local space $\mathscr{U}(I_j)$, which we again take to be polynomials of degree at most k, such that for all $v_h, w_h \in \mathscr{U}(I_j)$,

$$\begin{aligned} \int_{I_j} (u_h(x,t))_t\, v_h(x)\, dx + \int_{I_j} q_h(x,t)(v_h(x))_x\, dx - \hat{q}_h(\cdot,t)\, v_h \Big|_{x_{j-1/2}}^{x_{j+1/2}} &= 0 \\ \int_{I_j} q_h(x,t)\, w_h(x)\, dx + \int_{I_j} u_h(x,t)(w_h(x))_x\, dx - \hat{u}_h(\cdot,t)\, w_h \Big|_{x_{j-1/2}}^{x_{j+1/2}} &= 0 \end{aligned} \tag{5.4}$$

where, again for the lack of up-winding mechanism in a heat equation, one naturally first tries the central numerical fluxes:

$$\begin{aligned} \hat{u}_h(x_{j+1/2},t) &= \tfrac{1}{2}\,(u_h(x^-_{j+1/2},t) + u_h(x^+_{j+1/2},t)) \\ \hat{q}_h(x_{j+1/2},t) &= \tfrac{1}{2}\,(q_h(x^-_{j+1/2},t) + q_h(x^+_{j+1/2},t)) \end{aligned} \tag{5.5}$$

We emphasize that the above formulation of the discontinuous Galerkin scheme is only formally similar to that of the convection equation in Section 2. In fact, there is no time derivative in the second equation in (5.3) and it is *not* a hyperbolic problem even though it is written into a system form with only first derivatives. If we view the scheme (5.4) as a mixed finite element method then it lacks the usual sophisticated matching of the two solution spaces for u_h and q_h (the same space is used for both of them). "Common sense" in traditional finite elements would hint that scheme (5.4) has no chance to work. However, in 1997 Bassi and Rebay [18] were brave enough to try this method on the viscous terms in the Navier–Stokes equations and seemed to have obtained very good results. Motivated by their work, in 1998 Cockburn and Shu [50] analyzed this method and obtained conditions on the choice of the numerical fluxes $\hat{u}_h(x_{j+1/2},t)$ and $\hat{q}_h(x_{j+1/2},t)$ which guarantee stability, convergence and a sub-optimal error estimate of order k for piecewise polynomials of degree k. It turns out that the central numerical fluxes (5.5) used by Bassi and Rebay [18] do satisfy these conditions.

However, there are two problems associated with the choice of the central numerical fluxes in (5.5):

(i) It spreads to five cells when a local basis is chosen for u_h in the element I_j. After q_h is eliminated the scheme becomes

$$\begin{aligned} &\frac{d}{dt}(u_h)_j + \frac{1}{\Delta_j^2}\,(A(u_h)_{j-2} + B(u_h)_{j-1} + C(u_h)_j \\ &\quad + D(u_h)_{j+1} + E(u_h)_{j+2}) = 0 \end{aligned}$$

where $(u_h)_j$ is a small vector of length $k+1$ containing the coefficients of the solution u_h in the local basis inside the element I_j, and A, B, C D, E are $(k+1)\times(k+1)$ constant matrices which can be computed once and for all and stored at the beginning of the code. The stencil here is wider than that in (5.2).

(ii) The order of accuracy is one order lower for odd k. That is, for odd k the proof of the sub-optimal error estimate of order k is actually sharp.

Both problems can be cured by a suitable choice of the numerical fluxes, proposed in Cockburn and Shu [50]:

$$\hat{u}_h(x_{j+1/2}, t) = u_h(x^-_{j+1/2}, t), \qquad \hat{q}_h(x_{j+1/2}, t) = q_h(x^+_{j+1/2}, t) \tag{5.6}$$

that is, we alternatively take the left and right limits for the numerical fluxes in u_h and q_h (we could of course also take the pair $u_h(x^+_{j+1/2}, t)$ and $q_h(x^-_{j+1/2}, t)$ as the fluxes). Notice that the evaluation of (5.6) is simpler than that of the central fluxes in (5.5). Moreover, since the auxiliary variable q_h can be readily eliminated element-by-element, the actual scheme for u_h takes the form of (5.2) (of course with different constant matrices A, B and C) when a local basis is chosen. Hence the computational cost and storage requirement of scheme (5.4) with the numerical fluxes (5.6) is the same as that of the inconsistent scheme (5.1), even though we now have nominally an additional auxiliary variable q_h!

To illustrate the convergence properties of the scheme, we show in Table 5.2 the L^2 and L^∞ errors and numerically observed orders of accuracy,

Table 5.2. L^2 and L^∞ Errors and Orders of Accuracy for the Local Discontinuous Galerkin Method (5.4) with Fluxes (5.6) Applied to the Heat Equation $u_t = u_{xx}$ with an Initial Condition $u(x, 0) = \sin(x)$, $T = 0.8$. Third Order Runge–Kutta in Time

	$k=1$				$k=2$			
h	L^2 error	order	L^∞ error	order	L^2 error	order	L^∞ error	order
$2\pi/20$, u	1.92E-03	—	7.34E-03	—	4.87E-05	—	2.30E-04	—
$2\pi/20$, q	1.93E-03	—	7.33E-03	—	4.87E-05	—	2.30E-04	—
$2\pi/40$, u	4.81E-04	2.00	1.84E-03	1.99	6.08E-06	3.00	2.90E-05	2.99
$2\pi/40$, q	4.81E-04	2.00	1.84E-03	1.99	6.08E-06	3.00	2.90E-05	2.99
$2\pi/80$, u	1.20E-04	2.00	4.62E-04	2.00	7.60E-07	3.00	3.63E-06	3.00
$2\pi/80$, q	1.20E-04	2.00	4.62E-04	2.00	7.60E-07	3.00	3.63E-06	3.00
$2\pi/160$, u	3.00E-05	2.00	1.15E-04	2.00	9.50E-08	3.00	4.53E-07	3.00
$2\pi/160$, q	3.00E-05	2.00	1.15E-04	2.00	9.50E-08	3.00	4.53E-07	3.00

for both u_h and q_h, for the two cases $k=1$ and 2 (piecewise linear and piecewise quadratic cases) to $T=0.8$. Clearly $(k+1)$th order of accuracy is achieved for both odd and even k and also the same order of accuracy is achieved for q_h which approximates u_x; this orders of convergence have recently been proven by Castillo *et al.* [33]. We thus obtain the advantage of mixed finite element methods in approximating the derivatives of the exact solution to the same order of accuracy as the solution themselves, without additional storage or computational costs for the auxiliary variable q_h.

Another possible modification to the inconsistent scheme (5.1) is given by Baumann and Oden [22]; see also Oden *et al.* [92]. Basically, extra boundary terms were added to the element boundaries such that, when one takes $v=u$ and sums over all cells, the boundary contribution disappears and one gets a nice L^2 norm stability control. The scheme now becomes: find u_h whose restriction to each element I_j is, for each t, an element of the local space $\mathscr{U}(I_j)$, which we again take to be polynomials of degree at most k, such that for all $v_h \in \mathscr{U}(I_j)$,

$$\int_{I_j} (u_h(x,t))_t\, v_h(x)\, dx + \int_{I_j} (u_h(x,t))_x\, (v_h(x))_x\, dx - \widehat{(u_h)_x}\,(\cdot,t)\, v_h \Big|_{x_{j-1/2}}^{x_{j+1/2}} - \tfrac{1}{2}(v_h)_x\,(x_{j+1/2}^-)[u_h](x_{j+1/2},t) - \tfrac{1}{2}(v_h)_x\,(x_{j-1/2}^+)[u_h](x_{j-1/2},t) = 0 \tag{5.7}$$

where

$$[w] \equiv w^+ - w^-$$

denotes the jump of the function w at the interface and, again for the lack of upwinding mechanism in a heat equation, one naturally takes a central numerical flux

$$\widehat{(u_h)_x}\,(x_{j+1/2},t) = \tfrac{1}{2}\left((u_h)_x)(x_{j+1/2}^-,t) + (u_h)_x(x_{j+1/2}^+,t)\right)$$

For coding purposes, the equation (5.7) is the most convenient form; however it might be more illustrative if we rewrite (5.7) into a global form: find u_h whose restriction to each element I_j is, for each t, an element of the local space $\mathscr{U}(I_j)$, such that, for all v_h whose restriction to each element I_j is an element of $\mathscr{U}(I_j)$,

$$\int_0^{2\pi} (u_h(x,t))_t\, v_h(x)\, dx + \sum_{j=1}^{N} \left(\int_{I_j} (u_h(x,t))_x\, (v_h(x))_x\, dx + \widehat{(u_h)_x}\,(x_{j+1/2},t)[v_h](x_{j+1/2}) - \widehat{(v_h)_x}\,(x_{j+1/2})[u_h](x_{j+1/2},t) \right) = 0 \tag{5.8}$$

where the numerical flux for $(v_h)_x$ is also a central flux

$$\widehat{(v_h)_x}\,(x_{j+1/2}) = \tfrac{1}{2}\left((v_h)_x\,(x^-_{j+1/2}) + (v_h)_x\,(x^+_{j+1/2})\right)$$

The anti-symmetric nature of the boundary terms (which disappear when one takes $v_h = u_h$) is clearly seen in the global formulation (5.8).

We remark that once again we recover exactly the scheme in the form of (5.2) (of course with different constant matrices A, B and C) when a local basis is chosen. Hence the computational cost and storage requirement of scheme (5.7) is the same as that of the inconsistent scheme (5.1) or as that of the LDG method (5.4)–(5.6). There is no saving in the computational cost here over the method (5.4)–(5.6) even though the latter has nominally an additional auxiliary variable q_h. This statement is valid when a linear PDE is solved. For nonlinear problems the computational cost of the Baumann–Oden method (5.7) may be smaller than that of the LDG method (5.4)–(5.6).

To illustrate the performance of this method, we show in Table 5.3 the L^2 and L^∞ errors and numerically observed orders of accuracy, for the two cases $k = 1$ and 2 (piecewise linear and piecewise quadratic cases) to $T = 0.8$. Clearly $(k+1)$th order of accuracy is achieved for the odd $k = 1$ and kth order of accuracy is achieved for the even $k = 2$. Comparing with the results in Table 5.2 of the local discontinuous Galerkin method, we can see that, for the same mesh, the Baumann–Oden method (5.7) has larger errors than the local discontinuous Galerkin method (5.4)–(5.6) even for odd k where both are accurate of order $k+1$.

5.2. The LDG Methods for the Scalar One-Dimensional Case

We now turn our attention to more details about the LDG method, following the approach of Cockburn and Shu [50]. In this subsection, we

Table 5.3. L^2 and L^∞ Errors and Orders of Accuracy for the Baumann–Oden Discontinuous Galerkin Method (5.7) Applied to the Heat Equation $u_t = u_{xx}$ with an Initial Condition $u(x, 0) = \sin(x)$, $T = 0.8$. Third Order Runge–Kutta in Time

	$k=1$				$k=2$			
h	L^2 error	order	L^∞ error	order	L^2 error	order	L^∞ error	order
$2\pi/20$	6.40E-03	—	1.25E-02	—	4.00E-03	—	5.64E-03	—
$2\pi/40$	1.60E-03	2.00	3.14E-03	2.00	1.03E-03	1.95	1.46E-03	1.95
$2\pi/80$	4.00E-04	2.00	7.85E-04	2.00	2.61E-04	1.99	3.68E-04	1.99
$2\pi/160$	9.99E-05	2.00	1.96E-04	2.00	6.53E-05	2.00	9.23E-05	2.00

present and analyze the LDG methods for the following nonlinear model problem:

$$u_t + f(u)_x - (a(u)\, u_x)_x = 0, \qquad \text{in } (0,1)\times(0,T)$$
$$u(x,0) = u_0(x) \qquad \forall x \in (0,1)$$

where $a(u) \geqslant 0$, with periodic boundary conditions.

5.2.1. *General Formulation and Main Properties*

To define the LDG method, we set $b(u) = \sqrt{a(u)}$, introduce the new variable $q = b(u)\, u_x$, and rewrite our model problem as follows:

$$u_t + f(u)_x - (b(u)\, q)_x = 0 \qquad \text{in } (0,1)\times(0,T)$$
$$q - g(u)_x = 0 \qquad \text{in } (0,1)\times(0,T)$$
$$u(x,0) = u_0(x) \qquad \forall x \in (0,1)$$

where $g(u) = \int^u b(s)\, ds$. The LDG method for the above system is now obtained by simply discretizing the above system with the Discontinuous Galerkin method as follows.

As usual, for each time t, we take the restriction to the generic element I_j of the approximate solution $(u_h(\cdot,t), q_h(\cdot,t))$ in the space $\mathscr{U}(I_j)\times\mathscr{U}(I_j)$. The initial data $u_h(\cdot,0)$ on I_j is taken to be the L^2-projection of u_0 into $\mathscr{U}(I_j)$, and for $t>0$ the approximate solution is determined by requesting that, for all v_h and $r_h \in \mathscr{U}(I_j)$,

$$\int_{I_j} (u_h(x,t))_t\, v_h(x)\, dx - \int_{I_j} f(u_h)(x,t)(v_h)_x\,(x)\, dx + \hat{f}(u_h)(\cdot,t)\, v_h \Big|_{x_{j-1/2}}^{x_{j+1/2}}$$
$$+ \int_{I_j} b(u_h(x,t))\, q_h(t,x)(v_h)_x\,(x)\, dx - \widehat{bq}_h(\cdot,t)\, v_h \Big|_{x_{j-1/2}}^{x_{j+1/2}} = 0,$$

$$\int_{I_j} q_h(x,t)\, r_h(x)\, dx + \int_{I_j} g(u_h)(x,t)(r_h)_x\,(x)\, dx - \hat{g}_h(\cdot,t)\, r_h \Big|_{x_{j-1/2}}^{x_{j+1/2}} = 0$$

The key to the success of the LDG method is the choice of the numerical fluxes. The numerical flux $\hat{f}(u_h)$ is taken *exactly* as in the case of the scalar hyperbolic conservation of Section 2; in this way, the scheme remains

stable in the extreme case in which $a \equiv 0$. The numerical fluxes associated to the term modeling the diffusion are the following:

$$\widehat{bq}_h = \frac{[g(u_h)]}{[u_h]} \{q_h\} + C_{11}[u_h] + C_{12}[q_h], \qquad \hat{g}(u_h) = \{g(u_h)\} - C_{12}[u_h]$$

with $C_{11} \geqslant 0$. This completes the definition of the LDG space discretization. Let us emphasize the following points:

- The above numerical fluxes were devised *specifically* to ensure that the numerical scheme satisfies a discrete version of the the classical "energy" stability. More precisely, with the above numerical fluxes we get [50] that

$$E_T(u_h, q_h) \leqslant \tfrac{1}{2} \|u_0\|^2_{L^2(0,1)}$$

where the "energy" E_T is given by

$$E_T(u_h, q_h) = \tfrac{1}{2} \|u_h(T)\|^2_{L^2(0,1)} + \|q_h\|^2_{L^2((0,1)\times(0,T))} + \int_0^T \sum_{1 \leqslant j \leqslant N} \{\mathbb{C}_{11}[u_h]^2\}_{j+1/2}\, dt$$

and $\mathbb{C}_{11} = C_{11} + \frac{1}{[u_h]^2} \int_{u_h^-}^{u_h^+} (f(s) - \hat{f}(u_h^-, u_h^+))\, ds$.

This justifies the introduction of the function $b(u) = \sqrt{a(u)}$ and the definition $q = b(u)\, u_x$ which thus allows to obtain boundedness of the scheme for the non-linear problem.

Finally, note that this boundedness result assumes that we are performing the integration exactly. In practice, when strong non-linearities are present, high-order quadrature rules might be mandatory in order to maintain the boundedness of the scheme. An example of this situation can be found in the work of Lomtev *et al.* [91] who showed that, in order to produce high-quality approximations, over-integration of one or even two extra degrees of accuracy is necessary when steep gradients on the approximate solution appear near the boundary.

- In the linear case $f' \equiv c$ and $a(\cdot) \equiv a$, if we use polynomials of degree k, it was proven [50] that, if the exact solution is smooth enough, then

$$E_T(u - u_h, u_x - q_h) \leqslant Ch^k$$

where if $a = 0$, the constant C is of order $h^{1/2}$. This error estimate gives a sub-optimal order of convergence, but it is sharp for the LDG methods. Indeed, Bassi *et al.* [21] report an order of convergence of order $k+1$ for even values of k and of order k for odd values of k for a steady state,

purely elliptic problem for uniform grids and for c identically zero. The numerical results for a purely parabolic problem displayed in [50] support the same conclusions.

• On the other hand, for the special numerical flux (5.6), $c > 0$, and quite general boundary conditions, Castillo [31] and Cockburn *et al.* [33] showed that the order of convergence in the L^2-norm of both $u - u_h$ and $u_x - q_h$ is $k+1$; the h-version was studied in [31] while the hp-version of this method was studied in [33].

5.3. The LDG Methods for the Multi-Dimensional Case

In this subsection, we consider the LDG methods for the following convection-diffusion model problem

$$\begin{aligned} u_t + \nabla \cdot (f(u) - a(u)\nabla u) &= 0 && \text{in } (0,1)^d \times (0,T) \\ u(x,0) &= u_0(x) && \forall x \in (0,1)^d \end{aligned}$$

with periodic boundary conditions. Essentially, the one-dimensional case and the multi-dimensional case can be studied in exactly the same way. However, there are two important differences that deserve explicit discussion. The first is the treatment of the *matrix a* which is assumed to be *symmetric, semi-positive definite* and the introduction of the auxiliary variable q; and the second is the treatment of arbitrary meshes.

To define the LDG method, we first notice that, since the matrix $a(u)$ is assumed to be symmetric and semi-positive definite, there exists a symmetric matrix $b(u)$ such that $a = b^2$. This allows us to introduce the auxiliary variable $\mathbf{q} = b\nabla u$, and rewrite the model problem as follows:

$$\begin{aligned} u_t + \nabla \cdot f(u) - \nabla \cdot (b(u)\,\mathbf{q}) &= 0 && \text{in } (0,1)^d \times (0,T) \\ q_i &= \nabla \cdot \mathbf{g}_i(u) && \text{in } (0,1)^d \times (0,T), \quad 1 \leqslant i \leqslant d \\ u(x,0) &= u_0(x) && \forall x \in (0,1)^d \end{aligned}$$

where q_i is the ith component of the vector $\mathbf{q}$, and $\mathbf{g}_i(u)$ is the vector whose jth component is $\int^u b_{ji}(s)\,ds$. The LDG method is now obtained by discretizing the above equations by the Discontinuous Galerkin method.

Let $\mathcal{T}_h$ be a triangulation of the domain $(0,1)^d$. We seek an approximation $(u_h, \mathbf{q}_h)$ such that for each time t, its restriction to the element $K \in \mathcal{T}_h$ is in the space $\mathcal{U}(K) \times \mathcal{Q}^d(K)$. We take the restriction of $u_h(\cdot, 0)$ to K to be the L^2 projection of u_0 into $\mathcal{U}(K)$ and determine the approximate solution for $t > 0$ by imposing that, for $v_h \in \mathcal{U}(K)$ and $r_h \in \mathcal{Q}(K)$:

$$\int_K (u_h)_t\, v_h\, dx - \int_K f(u_h)\cdot\nabla v_h\, dx + \int_{\partial K} \widehat{f\cdot n_{\partial K}}(u_h)\, v_h\, ds$$

$$+ \int_K b(u_h)\, \mathbf{q}_h\cdot\nabla v_h\, dx - \int_{\partial K} \widehat{b\mathbf{q}}_h\cdot n_{\partial K} v_h\, ds = 0$$

$$\int_K q_{i,h} r_h\, dx + \int_K \mathbf{g}_{i,h}\cdot\nabla r_h\, dx - \int_{\partial K} \hat{\mathbf{g}}_{ih}\cdot n_{\partial K} r_h\, ds = 0$$

Finally, just like in the one-dimensional case, we take $\widehat{f\cdot n_{\partial K}}$ as we did in the purely convective case and

$$\widehat{b\mathbf{q}}_h = \sum_{i=1}^{d} \frac{[g_{i,h}]}{[u_h]}\{q_{i,h}\} + C_{11}[u_h] + \sum_{i=1}^{d} C_{1i}[q_{i,h}], \qquad \hat{g}(u_h) = \{g_{i_h}\} - C_{1i}[u_h]$$

For this method, we have properties similar to those obtained in the one-dimensional case:

- Energy stability:

$$E_T(u_h, \mathbf{q}_h) \leqslant \tfrac{1}{2}\|u_0\|^2_{L^2(0,1)}$$

where the "energy" E_T is given by

$$E_T(u_h, \mathbf{q}_h) = \tfrac{1}{2}\|u_h(T)\|^2_{L^2(0,1)} + \|q_h\|^2_{L^2((0,1)\times(0,T))} + \int_0^T \sum_{1\leqslant j\leqslant N} \{\mathbb{C}_{11}[u_h]^2\}_{j+1/2}\, dt$$

and $\mathbb{C}_{11} = C_{11} + \frac{1}{[u_h]^2}\int_{u_h^-}^{u_h^+} (f(s) - \hat{f}(u_h^-, u_h^+))\, ds$.

- In the linear case $f' \equiv c$ and $a(\cdot) \equiv a$, if we use polynomials of degree k, it was proven [50] that, if the exact solution is smooth enough, then

$$E_T(u - u_h, u_x - q_h) \leqslant Ch^k$$

where if $a = 0$, the constant C is of order $h^{1/2}$.

5.4. A Remark and Extension to Multi-Dimensional Systems

The main advantage of these methods is their extremely high parallelizability and their high-order accuracy which render them suitable for computations of convection-dominated flows. Indeed, although the LDG method have a large amount of degrees of freedom per element, and hence more computations per element are necessary, its extremely local domain of dependency allows a very efficient parallelization that by far compensates for the extra amount of local computations.

The LDG methods for multi-dimensional systems, like for example the compressible Navier–Stokes equations and the equations of the hydrodynamic model for semiconductor device simulation, can be easily defined by simply applying the procedure described for the multi-dimensional scalar case to each component of $\mathbf{u}$. In practice, especially for viscous terms which are not symmetric but still semi-positive definite, such as for the compressible Navier–Stokes equations, we can use $\mathbf{q} = (\partial_{x_1} u, \ldots, \partial_{x_d} u)$ as the auxiliary variables. Although with this choice, the L^2-stability result will not be available theoretically, this would not cause any problem in practical implementations, and does not seem to affect the excellent stability of the method in actual calculations.

5.5. Incompressible Navier–Stokes Equations

For the two dimensional incompressible Navier–Stokes equations in a vorticity-stream function formulation, Liu and Shu [88, 89] developed a numerical method based on a DG and LDG discretization for the vorticity equation including the viscous terms, a standard Poisson solver using continuous finite elements for the streamfunction, and a TVD Runge–Kutta time discretization. There is a natural matching between the two finite element spaces, since the normal component of the velocity field is *continuous* across element boundaries. This allows for a correct upwinding gluing in the discontinuous Galerkin framework, while still maintaining total energy conservation with no numerical dissipation and total enstrophy stability. In [88], a proof is given for L^2 stability, both in the total enstrophy (L^2-norm of the vorticity) and in the total energy (L^2-norm of the velocity), which does not depend on the regularity of the exact solutions. For smooth solutions error estimates are also obtained in [88]. Schemes with provable L^2-stability for both total energy and total enstrophy are very rare. Liu and Xin [90] used this nice stability property to show that the method in [88] converges with a vortex sheet initial data having only positive vorticity.

We present here one numerical example, taken from [89], a double shear layer problem. This is a popular benchmark problem for numerical methods of incompressible flows. The method in [88] is able to capture features of the solution with high gradients in a nice way. A higher order method is doing better in this respect than a lower order one. In Fig. 5.2, the simulation result with a uniform rectangular mesh of 256×256 cells with a piecewise quadratic method up to $T = 8$ is shown at the left for a very thin shear layer with a high Reynolds number $\mathrm{Re} = 70000/2\pi$. We notice that the numerical method is still stable in this case. A time history for energy and enstrophy shows that the physical viscosity is still dominating

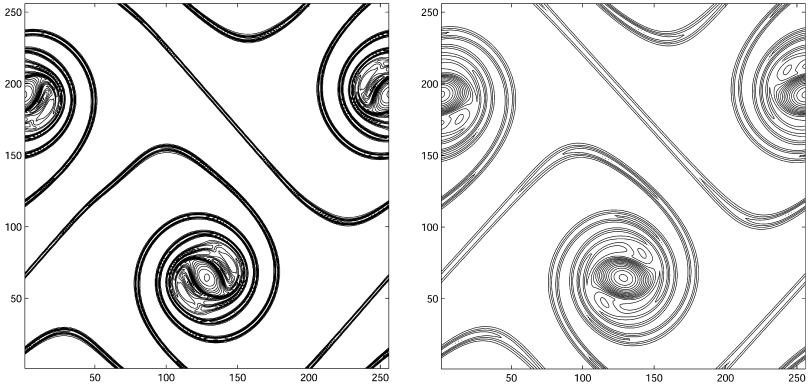

Fig. 5.2. Contour of vorticity ω at $T = 8$. 30 equally spaced contour lines between $\omega = -14.5$ and $\omega = 14.5$. Thin shear layer corresponding to $\rho = \pi/100$. $\mathrm{Re} = 70000/2\pi$. 256×256 cells with P2/Q2 scheme (left). As a comparison, we also plot the result of 256×256 cells with P1/Q1 scheme computation with $\mathrm{Re} = 20000/2\pi$ (right).

the numerics at such high Reynolds numbers, according to the decay of energy and enstrophy. This indicates that the built-in numerical viscosity of the methods is very small. For comparison, at the right of Fig. 5.2 the result of piecewise linear method for the same mesh is shown for a much lower Reynolds number $\mathrm{Re} = 20000/2\pi$.

6. HAMILTON–JACOBI AND NON-LINEAR SECOND-ORDER PARABOLIC EQUATIONS

In this section, we discuss the RKDG method for solving the following Hamilton–Jacobi (HJ) equations

$$\varphi_t + H(\nabla\varphi) = 0, \qquad \varphi(x, 0) = \varphi^0(x) \tag{6.1}$$

We also discuss RKDG methods for non-linear parabolic equations with viscosity solutions,

$$\varphi_t + F(D\varphi, D^2\varphi) = 0, \qquad \varphi(x, 0) = \varphi^0(x) \tag{6.2}$$

The weak solutions of the above problems are usually 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 [53].

We will mainly follow the 1999 paper by Hu and Shu [72], and the 2000 papers by Lepsky *et al.* [82] and Hu *et al.* [71]. We begin by considering the Hamilton–Jacobi equation. In many applications, practitioners are interested in solving (6.1) on an arbitrary triangulation in multiple space dimensions; there are several numerical schemes for this situation. Indeed, first order monotone type finite volume schemes and their second order extensions were studied by Abgrall in [1]. A second order ENO (essentially non-oscillatory) type finite volume scheme was developed by Lafon and Osher in [80]; see also the work of Augoula and Abgrall in [13]. However, higher order finite volume schemes face the problem of reconstruction on arbitrary triangulation, which is quite complicated.

It is well known that the Hamilton–Jacobi equation (6.1) is closely related to a conservation law (2.1), in fact in one space dimension $d = 1$ they are equivalent if one takes $\varphi = u_x$. It is thus not surprising that many successful numerical methods for the Hamilton–Jacobi equation (6.1) are adapted from those for the conservation law. Such examples include the high order finite difference ENO methods in Osher and Sethian [94], Osher and Shu [95], and WENO methods in Jiang and Peng [75]. However, it seems that such an adaptation is more difficult for unstructured meshes, especially for finite element methods which are usually based on integration by parts. The RKDG method we will discuss below is such an adaptation.

6.1. One Space Dimension

In one space dimension (6.1) becomes

$$\varphi_t + H(\varphi_x) = 0, \qquad \varphi(x, 0) = \varphi^0(x) \tag{6.3}$$

This is a relatively easy case because (6.3) is equivalent to the conservation law

$$u_t + H(u)_x = 0, \qquad u(x, 0) = u^0(x) \tag{6.4}$$

if we identify $u = \varphi_x$. If we take the local space $\mathscr{U}(I_j)$ to be the set of all polynomials of degree at most k and denote it by $P^k(I_j)$, then a kth order discontinuous Galerkin scheme for (6.3) can be defined as follows: find φ_h, whose restriction to $x \in I_j$ for each t is in $P^k(I_j)$, such that for all $v_h \in P^{k-1}(I_j)$,

$$\int_{I_j} (\varphi_h(x, t))_{xt}\, v_h(x)\, dx - \int_{I_j} H((\varphi_h(x, t))_x)\, v_h(x)\, dx$$

$$+ \hat{H}((\varphi_h(\cdot, t))_x)\, v_h \Big|_{x_{j-1/2}}^{x_{j+1/2}} = 0 \tag{6.5}$$

Here the numerical flux

$$\hat{H}((\varphi_h(x_{j+1/2}, t))_x) = \hat{H}((\varphi_h(x^-_{j+1/2}, t))_x, (\varphi_h(x^+_{j+1/2}, t)_x) \tag{6.6}$$

is again a monotone flux, i.e., $\hat{H}$ is non-decreasing in the first argument and non-increasing in the second, 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

$$\hat{H}(u^-, u^+) = H\left(\frac{u^- + u^+}{2}\right) - \frac{1}{2}\alpha\,(u^+ - u^-) \tag{6.7}$$

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 Section 2.1. Notice that the method described above is exactly the discontinuous Galerkin method for the conservation law equation (6.4) satisfied by the derivative $u = \varphi_x$. This only determines φ_h 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:

(i) By requiring that the residue has zero mean in each element I_j, i.e.,

$$\int_{I_j} ((\varphi_h(x, t))_t + H((\varphi_h(x, t))_x))\, dx = 0 \tag{6.8}$$

(ii) By using (6.8) to update only one (or a few) elements, e.g., the left-most element I_1, then use

$$\varphi_h(x_j, t) = \varphi_h(x_1, t) + \int_{x_1}^{x_j} (\varphi_h(x, t))_x\, dx \tag{6.9}$$

to determine the missing constant for the cell I_j.

We remark that, in the second approach, the recovered values of φ_h are dependent upon the choice of the starting point x_1. However this difference is on the level of the truncation errors and does not affect the order of accuracy. 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.

Concerning the stability of the method proposed above, using the cell entropy inequality (which implies L^2 stability) for the method of lines DG method for scalar nonlinear conservation laws in Jiang and Shu [74], we can easily obtain a uniform total variation bound for the numerical solution φ_h, see [72] for details. This is actually a rather strong stability result, as 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_h when $h \to 0$. However, this stability result is not strong enough to imply that the limit solution is the viscosity solution of (6.3).

Time discretization of (6.4) is again by the TVD Runge–Kutta methods discussed in Section 2.2.

6.2. Multiple Space Dimensions

Next we will discuss the case of multiple space dimensions, using the two-dimensional case to illustrate the ideas; the algorithm in more spatial dimensions is similar. This time, the scalar Hamilton–Jacobi equation

$$\varphi_t + H(\varphi_{x_1}, \varphi_{x_2}) = 0, \qquad \varphi(x, 0) = \varphi^0(x) \tag{6.10}$$

is in some sense equivalent to the following *system* of conservation laws

$$u_t + H(u, v)_{x_1} = 0, \quad v_t + H(u, v)_{x_2} = 0, \qquad (u(x, 0), v(x, 0)) = (u^0(x), v^0(x)) \tag{6.11}$$

if we identify

$$(u, v) = \nabla\varphi \tag{6.12}$$

For example, a vanishing viscosity solution of (6.10) corresponds, via (6.12), to a vanishing viscosity solution of (6.11), and vice versa [76]. However, (6.11) 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_h as our solution variable (a polynomial) and take its derivatives as u_h and v_h. This is the main thrust of the discontinuous Galerkin method developed in [72] and later in [82] and [71] for solving multi-dimensional Hamilton–Jacobi equations.

The solution procedure, for a DG spatial discretization and Euler forward time stepping (TVD Runge–Kutta time stepping is just a combination of several Euler forward steps), consists of the following:

- Use the DG discretization for the hyperbolic system (6.11) with a local Lax–Friedrichs flux (see Section 3), taking (u_h, v_h) at time level n by (6.12), and take a forward Euler time step to get a provisional value of (u_h, v_h) at time level $n+1$;
- Determine $\nabla\varphi_h$ at time level $n+1$ by a least square procedure:

$$\|\nabla\varphi_h - (u_h, v_h)\|_{L^2(K)} = \min_{\psi_h \in P^k(K)} \|\nabla\psi_h - (u_h, v_h)\|_{L^2(K)} \tag{6.13}$$

- The missing constant can again be obtained in one of the following two ways:

(i) By requiring that the residue has zero mean in each element K, i.e.,

$$\int_K ((\varphi_h(x,t))_t + H(\nabla\varphi_h(x,t)))\, dx = 0 \tag{6.14}$$

(ii) By using (7.14) to update only one (or a few) elements, e.g., the corner element(s), then use

$$\varphi_h(B,t) = \varphi_h(A,t) + \int_A^B (\varphi_h)_{x_1}\, dx_1 + (\varphi_h)_{x_2}\, dx_2 \tag{6.15}$$

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_h 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. It is important here that φ_h is a single function and u_h and v_h are just its derivatives. Otherwise the second approach would be questionable in effectively recovering φ_h.

It can be proven [82] that the least square procedure (6.13) maintains the mean values of u and v (i.e., the mean value of $(\varphi_h)_{x_1}$ equals that of u and the the mean value of $(\varphi_h)_{x_2}$ equals that of v) and does not increase the L^2-norm of $\nabla\varphi_h$ (i.e., the L^2-norm of $\nabla\varphi_h$ is no bigger than the sum of the L^2-norms of u and v). Thus it does not destroy the nice stability property of the RKDG method.

6.3. The Non-Linear Second-Order Parabolic Problem

The extension of the RKDG method for Hamilton–Jacobi equations to non-linear second-order parabolic equations is analogous to the extension of the RKDG method for hyperbolic systems to convection-diffusion systems.

So, if we are to discretize

$$\varphi_t + F(\varphi_x, \varphi_y, \varphi_{xx}, \varphi_{xy}, \varphi_{yy}) = 0, \qquad \varphi(x, 0) = \varphi_0(x)$$

we simply write a problem for $(u, v, p, q, r) = (\varphi_x, \varphi_y, \varphi_{xx}, \varphi_{xy}, \varphi_{yy})$, namely,

$$u_t + F(u, v, p, q, r)_x = 0, \qquad v_t + H(u, v, p, q, r)_y = 0,$$
$$p - u_x = 0, \qquad q - u_y = 0, \qquad r - v_y = 0,$$
$$u(x, y, 0) = (\varphi_0)_x\,(x, y), \qquad v(x, y, 0) = (\varphi_0)_y\,(x, y)$$

Now, we simply apply the RKDG method to the above problem to produce an approximation $(u_h, v_h, p_h, q_h, r_h)$ to (u, v, p, q, r). Then, we define the approximation φ_h to φ by solving the problem as was done for the Hamilton–Jacobi equations.

6.4. Numerical Examples

We will show two numerical examples here to illustrate the RKDG method for two dimensional Hamilton–Jacobi equations. More examples can be found in [72], [82] and [71].

The first example is the problem of a propagating surface on the unit disk:

$$\begin{cases} \varphi_t - (1 - \varepsilon K)\sqrt{1 + \varphi_{x_1}^2 + \varphi_{x_2}^2} = 0 \\ \varphi(x, 0) = \sin\left(\dfrac{\pi(x_1^2 + x_2^2)}{2}\right) \end{cases} \tag{6.16}$$

where K is the mean curvature defined by

$$K = -\frac{\varphi_{x_1x_1}(1 + \varphi_{x_2}^2) - 2\varphi_{x_1x_2}\varphi_{x_1}\varphi_{x_2} + \varphi_{x_2x_2}(1 + \varphi_{x_1}^2)}{(1 + \varphi_{x_1}^2 + \varphi_{x_2}^2)^{\frac{3}{2}}} \tag{6.17}$$

and ε is a small constant, subject to a Neumann type boundary condition $\nabla\varphi = 0$.

This problem (defined on a rectangle rather than on a circle) was studied in [94] by using a finite difference ENO schemes. It is difficult to use rectangular meshes when the domain is a circle. Instead, we use the triangulation shown in Fig. 6.1. Notice that the mesh has been refined near the center of the domain where the solution develops discontinuous derivatives (for the $\varepsilon = 0$ case). There are 1792 triangles and 922 nodes in this triangulation. The solution with $\varepsilon = 0$ is displayed in Fig. 6.2, and that with $\varepsilon = 0.1$ is displayed in Fig. 6.3. Notice that the solution at $t = 0$ is shifted downward by 0.2 to show the detail of the solution at later time.

Next we present a problem from computer vision [101]:

$$\begin{cases} \varphi_t + I(x)\sqrt{1+\varphi_{x_1}^2+\varphi_{x_2}^2} - 1 = 0, & -1 < x_1 < 1, -1 < x_2 < 1 \\ \varphi(x, 0) = 0 \end{cases} \tag{6.18}$$

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 $I(x) = 1$. Conditions must be prescribed at such points. Since our method is a finite element method, we need to prescribe suitable conditions at the corresponding elements. We take

$$I(x) = 1/\sqrt{1+(1-|x_1|)^2+(1-|x_2|)^2}$$

The exact steady solution is $\varphi(x, \infty) = (1-|x_1|)(1-|x_2|)$. We use a uniform rectangular mesh of 40×40 elements and impose the exact boundary conditions for $u = \varphi_{x_1}$, $v = \varphi_{x_2}$ from the above exact steady solution, and take the exact value at one point (the lower left corner) to recover φ_h. The results for P^2 and P^3 are presented in Fig. 6.4, while Fig. 6.5 contains the history of iterations to the steady state, indicating a nice convergence to machine zero of the numerical residue.

7. ONGOING WORK AND OPEN PROBLEMS

In this section, we bring to the attention of the reader some of the problems which we feel would be interesting to consider to further develop the RKDG methods.

7.1. Generalized Slope Limiters

As we have seen, an important component of the RKDG method for transient non-linear hyperbolic systems is the generalized slope limiter.

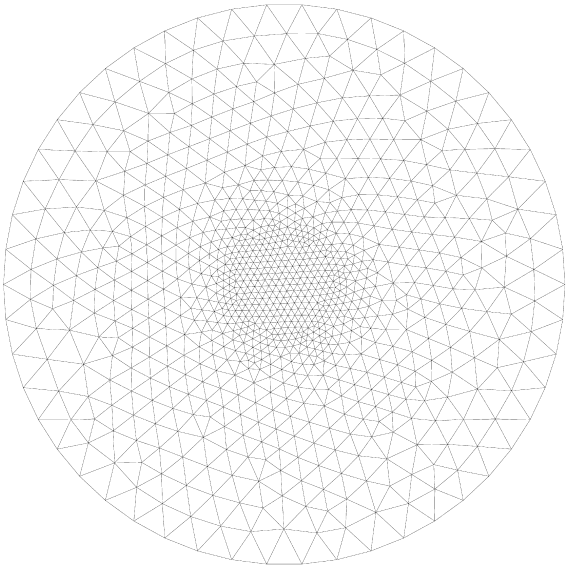

Fig. 6.1. Triangulation for the propagating surfaces on a disk.

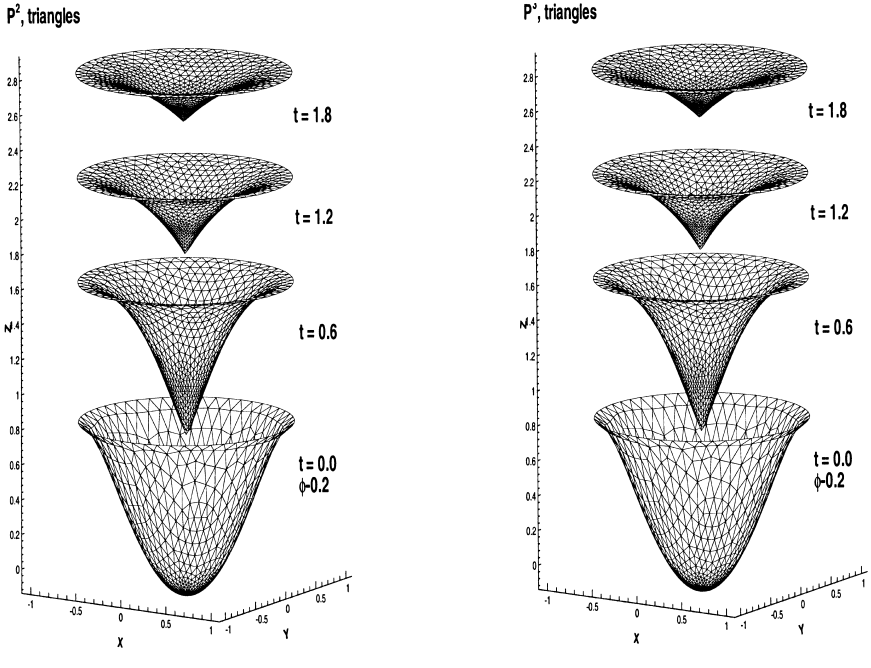

Fig. 6.2. Propagating surfaces on a disk, triangular mesh, $\varepsilon = 0$.

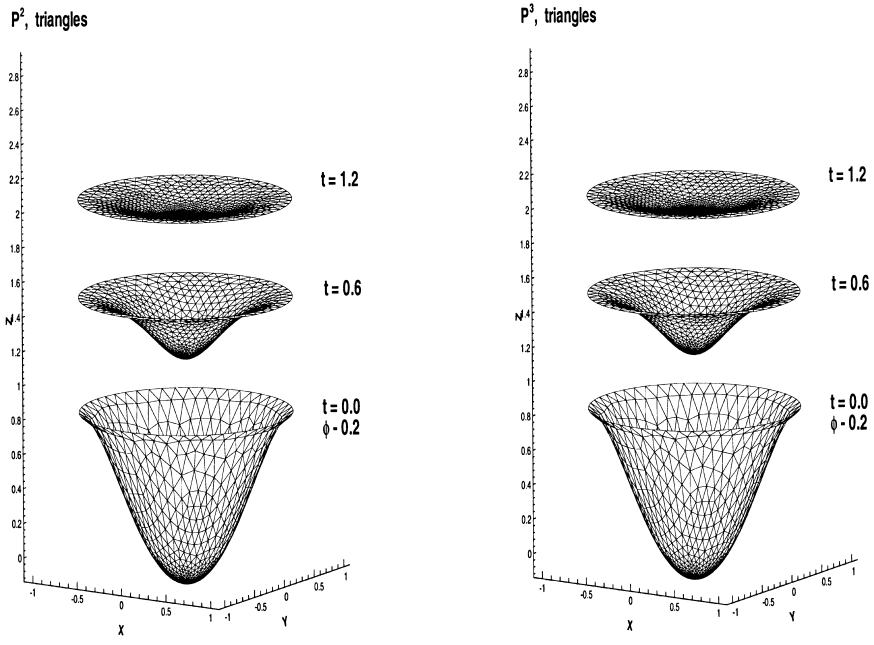

Fig. 6.3. Propagating surfaces on a disk, triangular mesh, $\varepsilon = 0.1$.

Although we have shown a generalized slope limiter that works very well, further research on limiters is very desirable.

First of all, let us emphasize that the limiter is *not* necessary for linear problems, but is *indispensable* for non-linear problems. This has been shown in the short essay by Cockburn [39] for non-linear scalar conservation laws, where it is pointed out that the limiter plays a role similar to that played by the shock-capturing terms of the streamline-diffusion method.

Further research is needed to find an efficient way to estimate the parameter M by means of which the limiter $\Lambda\Pi_{h,M}$ maintains the accuracy of the scheme at critical points. An ideal solution would be if this could be

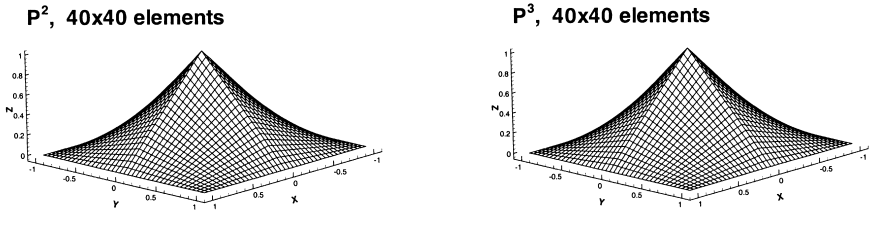

Fig. 6.4. Computer vision problem.

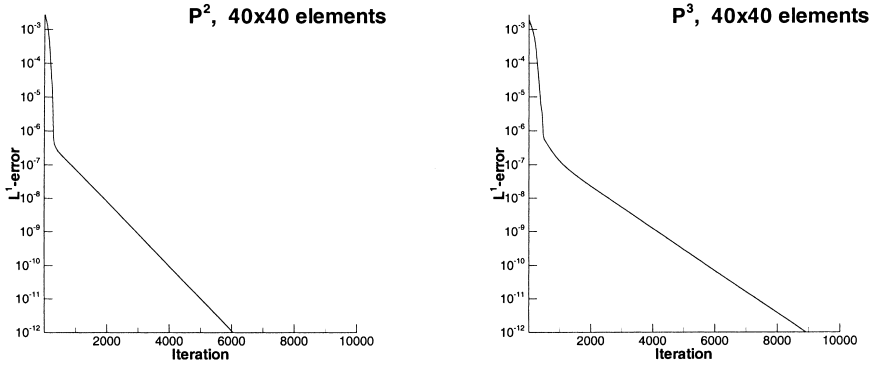

Fig. 6.5. Computer vision problem, history of iterations.

achieved in terms of the approximate solution only, and if this could be easily applied to general hyperbolic systems.

Another challenging problem is how to devise a limiter that is free from such a parameter. The limiter of Biswas *et al.* [26] has such a property. However, no stability results have been proven for this limiter and it only works in Cartesian grids with tensor product polynomial approximations.

7.2. Time-Stepping Techniques

In order to be able to perform adaptivity while maintaining the high parallelizability of the DG methods, new high-order accurate time-stepping methods would have to be created which could use different time steps at different locations. The space-time DG methods could be used to this effect, but they tend to be rather difficult to code.

Another possibility is to extend to high-order accurate schemes the approach used in 1995 by Dawson [54] to devise a first-order accurate, *conservative* variable time-stepping scheme; a significant achievement in this direction is the recent paper by Dawson and Kirby [55] who discovered how to obtain second-order accurate schemes of this type.

Non-conservative time-stepping methods can also lead to efficient time discretizations, but one has to be very careful to exert a tight control on the loss of mass, especially near the discontinuities; an example of this technique is the local time stepping introduced in 1997 by Flaherty *et al.* [61].

7.3. Enhanced Accuracy by Post-Processing

It is advantageous to know how to locally post-process the approximate solution in order to obtain a better approximation; this is particularly

true in the framework of a posteriori error estimation and adaptive algorithms. For DG methods, this has been done, so far, in two different ways: By finding super-convergence points and by a local convolution.

In 1994, Biswas *et al.* [26] gathered numerical evidence that, when rectangular elements are used, the approximate solution of the DG method super-converges at the Gauss-Radau points. This fact was exploited for adaptivity purposes then and recently proven by Adjerid *et al.* [4]; see also the papers by Adjerid, Aiffa and Flaherty [3] and [2]. Further understanding of this phenomena is very important.

Also recently, Cockburn *et al.* [47] showed that under some circumstances, it is possible to locally post-process the approximate solution of linear hyperbolic systems given by the DG space approximation and recover an order of accuracy of $2k+1$ instead of the expected order of $k+1/2$. The idea is based on a technique introduced by Bramble and Schatz [28] in the framework of finite element methods for linear elliptic problems and it requires locally uniform grids.

Let us illustrate the above result by showing some numerical results reported in [47]. We consider the model problem:

$$u_t + u_x = 0, \quad \text{in } (0,1)\times(0,T), \qquad u(x,0) = \sin(2\pi x), \quad \text{for } x \in (0,1)$$

subject to periodic boundary conditions. We denote by u_h the approximate solution obtained by using the DG method with piecewise polynomials of degree k over *uniform* grids of spacing h. We also consider the post-processed approximation $u_h^\star = K_h^k \star u_h$, where the convolution kernel K_h^k is a linear combination of B-splines that has support in $[-h(k-1/2), h(k+1/2)]$ and reproduces polynomials of degree $2k+1$ by convolution.

In Fig. 7.1 we display, for $T = 0.1$ and $h = 1/10$ and $h = 1/20$, the errors $x \mapsto u(T,x) - u_h(T,x)$ and $x \mapsto u(T,x) - u_h^\star(T,x)$ for $k = 1$ and in Fig. 7.2 for $k = 2$. Note how the oscillations in the error $x \mapsto u(T,x) - u_h(T,x)$ typical of finite element methods are remarkably reduced after the post-processing. Finally, in Table 7.1, we can see that the post-processed approximate solution converges with order $2k+1$, as claimed.

7.4. Application to Non-Convection-Diffusion Problems

So far, the main application of RKDG methods has been to compressible fluid flow, but there are many other problems on which a DG method could be very advantageous. For example, when applied to *linear* problems like Maxwell's equations, the mass matrix can be made to be the identity regardless of the polynomial degree and, moreover, the slope limiter often does not need to be used to guarantee stability. Also, applications of DG

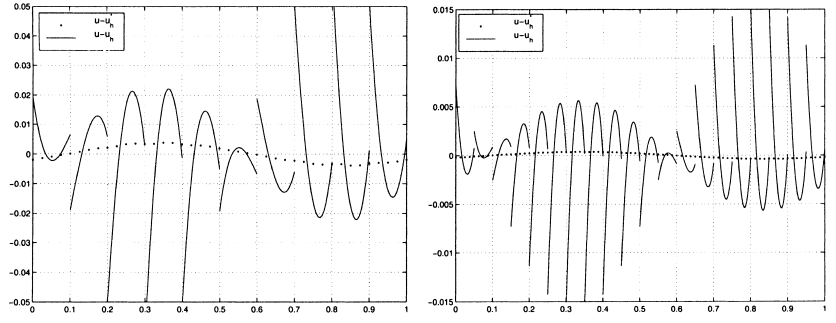

Fig. 7.1. The errors $u-u_h$ (solid line) and $u-u_h^\star$ (dots) at $T=0.1$ for $h=1/10$ (left) and $h=1/20$ (right). The function u is the smooth exact solution, u_h is the approximation given by the DG method with polynomials of degree one, and $u_h^\star$ is the post-processed solution.

methods to other situations like wave propagation phenomena in general, linear and non-linear solid mechanics, and non-linear equations like the Korteweg-de-Vries equations, just to name a few, has only began.

7.5. Relation of the LDG Method with Other Methods

A deep study of the relation of the DG methods to already existing methods could prove to be very illuminating not only from a theoretical point of view but also from a practical point of view since then it would be known for what situations it is more advantageous to use one method or the other. A first effort in this direction has been done by Arnold *et al.* [11] who established a unified framework to study and compare the LDG method and almost all other DG methods for elliptic problems. This theoretical study should be followed by a most needed computational study.

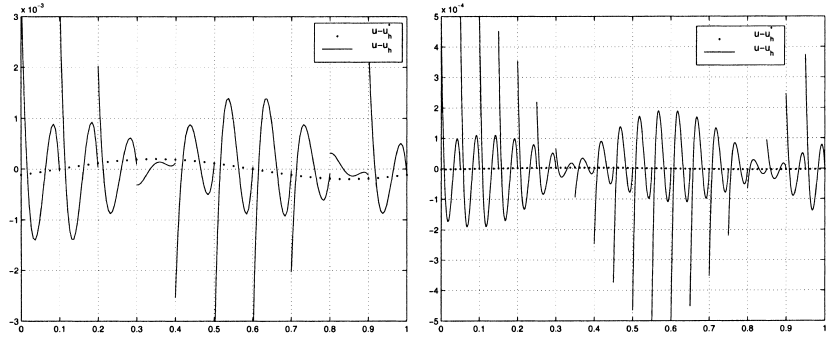

Fig. 7.2. The errors $u-u_h$ (solid line) and $u-u_h^\star$ (dots) at $T=0.1$ for $h=1/10$ (left) and $h=1/20$ (right). The function u is the smooth exact solution, u_h is the approximation given by the DG method with polynomials of degree two, and $u_h^\star$ is the post-processed solution.

Table 7.1. $u_t + u_x = 0$, Smooth Solution

	Before postprocessing				After postprocessing			
mesh	L^2 error	order	L^∞ error	order	L^2 error	order	L^∞ error	order
				P^1				
10	3.29E-02	—	5.81E-02	—	3.01E-02	—	4.22E-02	—
20	5.63E-03	2.55	1.06E-02	2.45	3.84E-03	2.97	5.44E-03	2.96
40	1.16E-03	2.28	2.89E-03	1.88	4.79E-04	3.00	6.78E-04	3.01
80	2.72E-04	2.09	8.08E-04	1.84	5.97E-05	3.00	8.45E-05	3.00
160	6.68E-05	2.03	2.13E-04	1.93	7.45E-06	3.00	1.05E-05	3.00
320	1.66E-05	2.01	5.45E-05	1.96	9.30E-07	3.00	1.32E-06	3.00
				P^2				
10	8.63E-04	—	2.86E-03	—	2.52E-04	—	3.57E-04	
20	1.07E-04	3.01	3.69E-04	2.95	5.96E-06	5.40	8.41E-06	5.41
40	1.34E-05	3.00	4.63E-05	3.00	1.53E-07	5.29	2.16E-07	5.28
80	1.67E-06	3.00	5.78E-06	3.00	4.22E-09	5.18	5.97E-09	5.18
160	2.09E-07	3.00	7.23E-07	3.00	1.27E-10	5.06	1.80E-10	5.06
				P^3				
10	3.30E-05	—	9.59E-05	—	1.64E-05	—	2.31E-05	—
20	2.06E-06	4.00	6.07E-06	3.98	7.07E-08	7.85	1.00E-07	7.85
40	1.29E-07	4.00	3.80E-07	4.00	2.91E-10	7.92	4.15E-10	7.91
50	5.29E-08	4.00	1.56E-07	4.00	5.03E-11	7.87	7.24E-11	7.83
				P^4				
10	1.02E-06	—	2.30E-06	—	1.98E-06	—	2.81E-06	—
20	3.21E-08	5.00	7.30E-08	4.98	2.20E-09	9.82	3.11E-09	9.82
30	4.23E-09	5.00	9.66E-09	4.99	4.34E-11	9.68	6.66E-11	9.48

It would also be very interesting to understand how to couple DG methods with other methods. This is of great practical interest since in many practical situations, already existing methods (and codes!) work just fine in some parts of the domain but not in others where the use of a DG method could be indispensable. For example, in the framework of CFD, the practitioner might want to use the LDG only in a region in which the convection has a strong effect and use elsewhere another method that works well when diffusion dominates.

A significant effort in this direction has been recently done by Perugia and Schötzau [96] who showed how to couple the LDG method with the classical conforming finite element method for the model elliptic problem of the Laplacian operator. Their motivation comes from a problem involving rotating electrical machines which are triangulated independently of

each other; see [8] for details. In this instance, the LDG is used to deal with the hanging nodes that naturally arise in this problem.

The coupling is done as follows. The LDG method is applied on the domain Ω_{LDG} and the conforming method on $\Omega_C = \Omega \setminus \bar{\Omega}_{LDG}$. The coupling takes place at the common boundary of Ω_{LDG} and Ω_C which we denote by Γ. To define the LDG on Ω_{LDG}, the boundary Γ is considered to be a *Dirichlet* boundary on which the data is the value given by the trace of the conforming approximation on Γ. To define the conforming method on Ω_C, the boundary Γ is now considered to be a *Neumann* boundary on which data is given by the corresponding numerical flux of the LDG method.

Perugia and Schötzau [96] proved that when elements of degree k are used on each variable, optimal orders of convergence are achieved. Next, we display some of their numerical results. In Fig. 7.3, we show the grids used in the experiments; note that the domain Ω_{LDG} contains all the hanging nodes and shrinks towards them as the meshes are refined. In Table 7.2 we can see that the error in the energy semi-norm, namely, $\{\|\mathbf{q}-\mathbf{q}_h\|_\Omega^2 + \sum_{\text{edges } e} \int_e C_{11}[u-u_h]^2\}^{1/2}$, and the L^2-norm of $u-u_h$ converge with optimal order, as expected. This shows that the coupling of the LDG and the conforming method can be successfully carried out.

7.6. Efficient Steady State Solvers

One problem with following physical time to reach steady states for convection dominated convection diffusion problems using RKDG methods is that the CFL condition for L^2-stability severely restricts the allowable

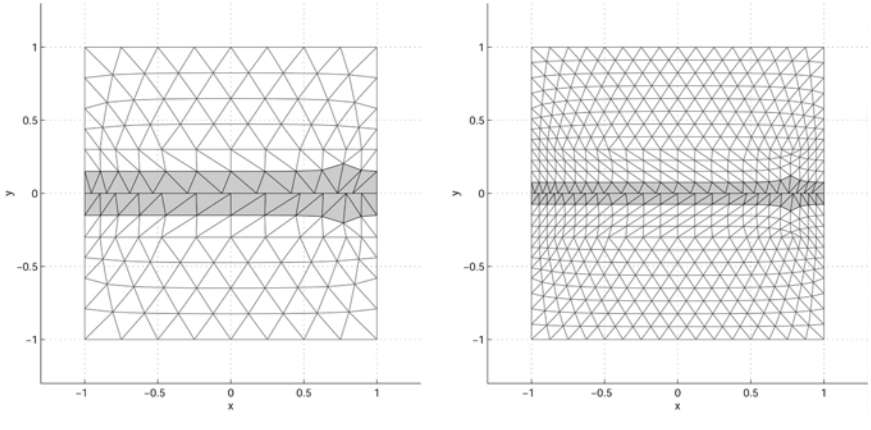

Fig. 7.3. Grids used in the numerical experiments: non–nested grids with hanging nodes on the line $y = 0$ (bottom). The domain Ω_{LDG} is shadowed.

Table 7.2. Errors and Orders of Convergence for the Coupling of the LDG and the Conforming Finite Element Method

	Energy-seminorm		L^2-norm of u	
reduction in h	error	order	error	order
—	4.7506e-1	—	6.5124e-2	–
2.0	2.5514e-1	1.8620	1.7317e-2	3.7607
2.0	1.3239e-1	1.9272	4.5057e-3	3.8434
2.0	6.7376e-2	1.9649	1.1496e-3	3.9194

time step Δt, making the marching in time rather slow. Various preconditioning and multigrid techniques would seem desirable here. A challenge is that one would not want to give up the extremely local property of the method which is responsible for its high parallel efficiency. Such techniques could also be useful for time dependent calculations through the introduction of a pseudo time.

To end this paper, let us emphasize that there are many aspects concerning the development of DG methods that we have not touched upon this review. Maybe the most important is the issue of adaptivity and parallelizability; see, for example, the papers on the subject that have appeared in [45]. As we have argued, the DG methods are ideally suited for adaptivity and parallelizability and might be the methods of choice for the use of adaptive strategies combined with load balancing techniques, not only in computational fluid dynamics but in a wide variety of problems of practical interest.

ACKNOWLEDGMENTS

B.C. was supported in part by NSF Grant DMS-9807491 and by the University of Minnesota Supercomputing Institute. C.-W.S. was supported in part by ARO Grant DAAD19-00-1-0405, NSF Grants DMS-9804985 and ECS-9906606, NASA Langley Grant NCC1-01035 and Contract NAS1-97046 while this author was in residence at ICASE, NASA Langley Research Center, Hampton, VA 23681-2199, and AFOSR Grant F49620-99-1-0077.

REFERENCES

1. R. Abgrall, (1996). Numerical discretization of the first-order Hamilton–Jacobi equations on triangular meshes. *Comm. Pure Appl. Math.* **49**, 1339–1377.
2. Adjerid, S., Aiffa, M., and Flaherty, J. E. (1998). Computational methods for singularly perturbed systems. In Cronin, J., and O'Malley, R. E. (eds.), *Singular Perturbation Concepts of Differential Equations*, AMS Proceedings of Symposia in Applied Mathematics, AMS.

3. Adjerid, S., Aiffa, M., and Flaherty, J. E. (1995). High-order finite element methods for singularly-perturbed elliptic and parabolic problems. *SIAM J. Appl. Math.* **55**, 520–543.
4. Adjerid, S., Flaherty, J. E., and Krivodonova, L. Superconvergence and a posteriori error estimation for continuous and discontinuous Galerkin methods applied to singularly perturbed parabolic and hyperbolic problems, in preparation.
5. Aizinger, V., Dawson, C. N., Cockburn, B., and Castillo, P. (2000). Local discontinuous Galerkin method for contaminant transport. *Advances in Water Resources* **24**, 73–87.
6. Allmaras, S. R. (1989). *A Coupled Euler/Navier–Stokes Algorithm for 2-D Unsteady Transonic Shock/Boundary-Layer Interaction*, Ph.D. thesis, Massachussetts Institute of Technology.
7. Allmaras, S. R., and Giles, M. B. (1987). *A Second Order Flux Split Scheme for the Unsteady 2-D Euler Equations on Arbitrary Meshes*, 8th. AIAA Computational Fluid Dynamic Conference, Honolulu, Hawai, June 9–11. *AIAA*, 87-1119-CP.
8. Alotto, P., Bertoni, A., Perugia, I., and Schötzau, D. (2000). Discontinuous finite element methods for the simulation of rotating electrical machines, *Proceedings of 9th International IGTE Symposium on Numerical Field Calculation in Electrical Engineering*, September 11–14, Graz, Austria.
9. Arnold, D. N. (1982). An interior penalty finite element method with discontinuous elements. *SIAM J. Numer. Anal.* **19**, 742–760.
10. Arnold, D. N., Brezzi, F., Cockburn, B., and Marini, D. Unified analysis of discontinuous Galerkin methods for elliptic problems. *SIAM J. Numer. Anal.*, to appear.
11. Arnold, D. N., Brezzi, F., Cockburn, B., and Marini, D. (2000). Discontinuous Galerkin methods for elliptic problems. In Cockburn, B., Karniadakis, G. E., and Shu, C.-W. (eds.), *Discontinuous Galerkin Methods. Theory, Computation and Applications*, Lecture Notes in Computational Science and Engineering, Vol. 11, Springer-Verlag, pp. 89–101.
12. Atkins H. L., and Shu, C.-W. (1998). Quadrature-free implementation of discontinuous Galerkin methods for hyperbolic equations. *AIAA J.* **36**, 775–782.
13. Augoula, S., and Abgrall, R. (2000). High order numerical discretization for Hamilton–Jacobi equations on triangular meshes. *J. Sci. Comput.* **15**, 197–229.
14. Babuška, I., and Zlámal, M. (1973). Nonconforming elements in the finite element method with penalty. *SIAM J. Numer. Anal.* **10**, 863–875.
15. Baker, G. A. (1977). Finite element methods for elliptic equations using nonconforming elements. *Math. Comp.* **31**, 45–59.
16. Baker, G. A., Jureidini, W. N., and Karakashian, O. A. (1990). Piecewise solenoidal vector fields and the Stokes problem. *SIAM J. Numer. Anal.* **27**, 1466–1485.
17. Bardos, C., LeRoux, A. Y., and Nédélec, J. C. (1979). First order quasilinear equations with boundary conditions. *Comm. in P.D.E.* **4**, 1017–1034.
18. Bassi, F., and S. Rebay, (1997). A high-order accurate discontinuous finite element method for the numerical solution of the compressible Navier–Stokes equations. *J. Comput. Phys.* **131**, 267–279.
19. Bassi, F., and S. Rebay, (1997). High-order accurate discontinuous finite element solution of the 2D Euler equations. *J. Comput. Phys.* **138**, 251–285.
20. Bassi, F., and S. Rebay, (2000). GMRES for discontinuous Galerkin solution of the compressible Navier–Stokes equations. In Cockburn, B., Karniadakis, G. E., and Shu, C.-W. (eds.), *Discontinuous Galerkin Methods. Theory, Computation and Applications*, Lecture Notes in Computational Science and Engineering, Vol. 11, Springer-Verlag, pp. 197–208.
21. Bassi, F., Rebay, S., Mariotti, G., Pedinotti, S., and Savini, M. (1997). A high-order accurate discontinuous finite element method for inviscid and viscous turbomachinery flows. In Decuypere, R., and Dibelius, G. (eds.), *2nd European Conference on Turbomachinery Fluid*

Dynamics and Thermodynamics (Antwerpen, Belgium), March 5–7, Technologisch Instituut, pp. 99–108.

22. Baumann, C. E., and Oden, J. T. (1999). A discontinuous *hp* finite element method for convection-diffusion problems. *Comput. Methods Appl. Mech. Engrg.* **175**, 311–341.
23. Bernardi, C., Maday, Y., and Patera, A. T. (1993). Domain decomposition by the mortar element method. In Kaper, H. G., and Garbey, M. (eds.), *Asymptotic and Numerical Methods for Partial Differential Equations with Critical Parameters*, Kluwer Academic Publishers, pp. 269–286.
24. Bernardi, C., Maday, Y., and Patera, A. T. (1994). A new nonconforming approach to domain decomposition: The mortar element method. In Brézis, H., and Lions, J. L. (eds.), *Nonlinear Partial Differential Equations and Their Applications, Collège de France Seminar, Volume XI*, Pitman Research Notes in Mathematics, No. 299, Pitman Advanced Publishing Program.
25. Bernardi, C., Debit, N., and Maday, Y. (1990). Coupling finite element and spectral methods: First results. *Math. Comp.* **54**, No. 189, 21–39.
26. Biswas, R., Devine, K. D., and Flaherty, J. (1994). Parallel, adaptive finite element methods for conservation laws. *Appl. Numer. Math.* **14**, 255–283.
27. Bourgeat, A., and Cockburn, B. (1989). The TVD-projection method for solving implicit numerical schemes for scalar conservation laws: A numerical study of a simple case. *SIAM J. Sci. Stat. Comput.* **10**, 253–273.
28. Bramble, J. H., and Schatz, A. H. (1977). Higher order local accuracy by averaging in the finite element method. *Math. Comp.* **31**, 94–111.
29. Brezzi, F., Manzini, G., Marini, D., Pietra, P., and Russo, A., (2000). Discontinuous Galerkin approximations for elliptic problems. *Numer. Methods Partial Differential Equations* **16**, 365–378.
30. Brezzi, F., Marini, D., Pietra, P., and Russo, A. (1999). Discontinuous finite elements for diffusion problems, Atti Convegno in onore di F. Brioschi (Milano 1997), Istituto Lombardo, Accademia di Scienze e Lettere, pp. 197–217.
31. Castillo, P. (2000). An optimal error estimate for the local discontinuous Galerkin method. In Cockburn, B., Karniadakis, G. E., and Shu, C.-W. (eds.), *Discontinuous Galerkin Methods. Theory, Computation and Applications*, Lecture Notes in Computational Science and Engineering, Vol. 11, Springer-Verlag, pp. 285–290.
32. Castillo, P., Cockburn, B., Perugia, I., and Schötzau, D., (2000). An a priori error analysis of the local discontinuous Galerkin method for elliptic problems. *SIAM J. Numer. Anal.* **38**, 1676–1706.
33. Castillo, P., Cockburn, B., Schötzau, D., and Schwab, C. An optimal a priori error estimate for the *hp*-version of the local discontinuous Galerkin method for convection-diffusion problems. *Math. Comp.*, to appear.
34. Chavent, G., and Cockburn, B. (1989). The local projection P^0P^1-discontinuous-Galerkin finite element method for scalar conservation laws. *RAIRO Modél. Math. Anal. Numér.* **23**, 565–592.
35. Chavent, G., and Salzano, G. (1982). A finite element method for the 1D water flooding problem with gravity. *J. Comput. Phys.* **45**, 307–344.
36. Chen, Z., Cockburn, B., Gardner, C., and Jerome, J. (1995). Quantum hydrodynamic simulation of hysteresis in the resonant tunneling diode. *J. Comput. Phys.* **117**, 274–280.
37. Chen, Z., Cockburn, B., Jerome, J., and Shu, C.-W. (1995). Mixed-RKDG finite element methods for the 2-D hydrodynamic model for semiconductor device simulation. *VLSI Design* **3**, 145–158.
38. Cockburn, B. (1999). Discontinuous Galerkin methods for convection-dominated problems. In Barth, T., and Deconink, H. (eds.), *High-Order Methods for Computational*

Physics, Lecture Notes in Computational Science and Engineering, Vol. 9, Springer-Verlag, pp. 69–224.
39. Cockburn, B. (2001). Devising discontinuous Galerkin methods for non-linear hyperbolic conservation laws. *J. Comput. Appl. Math.* **128**, 187–204.
40. Cockburn, B., and Gremaud, P. A. (1996). Error estimates for finite element methods for nonlinear conservation laws. *SIAM J. Numer. Anal.* **33**, 522–554.
41. Cockburn, B., Hou, S., and Shu, C.-W. (1990). TVB Runge–Kutta local projection discontinuous Galerkin finite element method for conservation laws IV: The multidimensional case. *Math. Comp.* **54**, 545–581.
42. Cockburn, B., Kanschat, G., Perugia, I., and Schötzau, D. (2001). Superconvergence of the local discontinuous Galerkin method for elliptic problems on Cartesian grids. *SIAM J. Numer. Anal.* **39**, 264–285.
43. Cockburn, B., Kanschat, G., Schötzau, D., and Schwab, C. Local discontinuous Galerkin methods for the Stokes system. *SIAM J. Numer. Anal.*, to appear.
44. Cockburn, B., Karniadakis, G. E., and Shu, C.-W. (2000). The development of discontinuous Galerkin methods. In Cockburn, B., Karniadakis, G. E., and Shu, C.-W. (eds.), *Discontinuous Galerkin Methods. Theory, Computation and Applications*, Lecture Notes in Computational Science and Engineering, Vol. 11, Springer-Verlag, pp. 3–50.
45. Cockburn, B., Karniadakis, G. E., and Shu, C.-W. (eds.) (2000). *Discontinuous Galerkin Methods. Theory, Computation and Applications*, Lecture Notes in Computational Science and Engineering, Vol. 11, Springer-Verlag.
46. Cockburn, B., Lin, S. Y., and Shu, C.-W. (1989). TVB Runge–Kutta local projection discontinuous Galerkin finite element method for conservation laws III: One dimensional systems. *J. Comput. Phys.* **84**, 90–113.
47. Cockburn, B., Luskin, M., Shu, C.-W., and Süli, E. Enhanced accuracy by post-processing for finite element methods for hyperbolic equations, *Math. Comp.*, to appear.
48. Cockburn, B., and Shu, C.-W. (1989). TVB Runge–Kutta local projection discontinuous Galerkin finite element method for scalar conservation laws II: General framework. *Math. Comp.* **52**, 411–435.
49. Cockburn, B., and Shu, C.-W. (1991). The Runge–Kutta local projection P^1-discontinuous Galerkin method for scalar conservation laws. *RAIRO Modél. Math. Anal. Numér.* **25**, 337–361.
50. Cockburn, B., and Shu, C.-W. (1998). The local discontinuous Galerkin method for time-dependent convection-diffusion systems. *SIAM J. Numer. Anal.* **35**, 2440–2463.
51. Cockburn, B., and Shu, C.-W. (1998). The Runge–Kutta discontinuous Galerkin finite element method for conservation laws V: Multidimensional systems. *J. Comput. Phys.* **141** (1998), 199–224.
52. Crandall, M., and Majda, A. (1980). Monotone difference approximations for scalar conservation laws. *Math. Comp.* **34**, 1–21.
53. Crandall, M. G., and Lions, P. L. (1983). Viscosity solutions of Hamilton–Jacobi equations. *Trans. Amer. Math. Soc.* **277**, 1–42.
54. Dawson, C. N. (1995). High resolution upwind-mixed finite element methods for advection-diffusion equations with variable time-stepping. *Numer. Methods Partial Differential Equations* **11**, 525–538.
55. Dawson, C. N., and Kirby, R. (2001). High resolution schemes for conservation laws with locally varying time steps. *SIAM J. Math. Anal.* **22**, 2256–2281.
56. Douglas, Jr., J., Darlow, B. L., Kendall, R. P., and Wheeler, M. F. (1979). Self-adaptive Galerkin methods for one-dimensional, two-phase immiscible flow, *AIME Fifth Simposium on Reservoir Simulation (Denver, Colorado)*, Society of Petroleum Engineers, pp. 65–72.

57. Douglas, Jr., J., and Dupont, T. (1976). *Interior Penalty Procedures for Elliptic and Parabolic Galerkin Methods*, Lecture Notes in Physics, Vol. 58, Springer-Verlag, Berlin.
58. Dubiner, M. (1991). Spectral methods on triangles and other domains. *J. Sci. Comp.* **6**, 345–390.
59. Falk, R. (2000). Analysis of finite element methods for linear hyperbolic problems. In Cockburn, B., Karniadakis, G. E., and Shu, C.-W. (eds.), *Discontinuous Galerkin Methods. Theory, Computation and Applications*, Lecture Notes in Computational Science and Engineering, Vol. 11, Springer-Verlag, pp. 103–112.
60. Feng, X., and Karakashian, O. A. Two-level non-overlapping schwarz methods for a discontinuous Galerkin method. *SIAM J. Numer. Anal.*, to appear.
61. Flaherty, J. E., Loy, R. M., Shephard, M. S., Szymanski, B. K., Teresco, J. D., and Ziantz, L. H. (1997). Adaptive local refinement with octree load-balancing for the parallel solution of three-dimensional conservation laws. *J. Parallel and Dist. Comput.* **47**, 139–152.
62. Gopalakrishnan, J., and Kanshat, G. A multilevel discontinuous Galerkin method. *Numer. Math.*, to appear.
63. Gottlieb, S., and Shu, C.-W. (1998). Total variation diminishing Runge–Kutta schemes. *Math. Comp.* **67**, 73–85.
64. Gottlieb, S., Shu, C.-W., and Tadmor, E. (2001). Strong stability preserving high order time discretization methods. *SIAM Rev.* **43**, 89–112.
65. Halt, D. W. (1992). *A Compact Higher Order Euler Solver for Unstructured Grids*, Ph.D. thesis, Washington University.
66. Halt, D. W., and Agarwall, R. K. (1991). A compact higher order characteristic-based Euler solver for unstructured grids. *AIAA*, 91-3234.
67. Halt, D. W., and Agarwall, R. K. (1992). A compact higher order Euler solver for unstructured grids with curved boundaries. *AIAA*, 92-2696.
68. Harten, A. (1983). High resolution schemes for hyperbolic conservation laws. *J. Comput. Phys.* **49**, 357–393.
69. Harten, A., Hyman, J. M., and Lax, P. D. (1976). On finite difference approximations and entropy conditions for shocks. *Comm. Pure and Appl. Math.* **29**, 297–322.
70. Houston, P., Schwab, C., and Süli, E. (2000). Stabilized *hp*-finite element methods for hyperbolic problems. *SIAM J. Numer. Anal.* **37**, 1618–1643.
71. Hu, C., Lepsky, O., and Shu, C.-W. (2000). The effect of the lest square procedure for discontinuous Galerkin methods for Hamilton–Jacobi equations. In Cockburn, B., Karniadakis, G. E., and Shu, C.-W. (eds.), *Discontinuous Galerkin Methods. Theory, Computation and Applications*, Lecture Notes in Computational Science and Engineering, Vol. 11, Springer-Verlag, pp. 343–348.
72. Hu, C., and Shu, C.-W. (1999). A discontinuous Galerkin finite element method for Hamilton–Jacobi equations. *SIAM J. Sci. Comput.* **21**, 666–690.
73. Jaffré, J., Johnson, C., and Szepessy, A. (1995). Convergence of the discontinuous Galerkin finite element method for hyperbolic conservation laws. *Math. Models Methods Appl. Sci.* **5**, 367–386.
74. Jiang, G., and Shu, C.-W. (1994). On a cell entropy inequality for discontinuous Galerkin methods. *Math. Comp.* **62**, 531–538.
75. Jiang, G.-S., and Peng, D.-P. (2000). Weighted ENO schemes for Hamilton–Jacobi equations. *SIAM J. Sci. Comput.* **21**, 2126–2143.
76. Jin, S., and Xin, Z.-P. (1998). Numerical passage from systems of conservation laws to Hamilton–Jacobi equation. *SIAM J. Numer. Anal.* **35**, 2385–2404.
77. Johnson, C., and Pitkäranta, J. (1986). An analysis of the discontinuous Galerkin method for a scalar hyperbolic equation. *Math. Comp.* **46**, 1–26.

78. Karniadakis, G. E., and Sherwin, S. J. (1999). *Spectral/hp Element Methods in CFD*, Oxford University Press.
79. Kuznetsov, N. N. (1976). Accuracy of some approximate methods for computing the weak solutions of a first-order quasi-linear equation. *USSR Comp. Math. and Math. Phys.* **16**, 105–119.
80. Lafon, F., and Osher, S. (1996). High-order 2-dimensional nonoscillatory methods for solving Hamilton–Jacobi scalar equations. *J. Comput. Phys.* **123**, 235–253.
81. Lasser, C., and Toselli, A. (2000). *An Overlapping Domain Decomposition Preconditioner for a Class of Discontinuous Galerkin Approximations of Advection-Diffusion Problems*, Tech. Report 2000-12, Seminar für Angewandte Mathematik, ETH Zürich.
82. Lepsky, O., Hu, C., and Shu, C.-W. (2000). Analysis of the discontinuous Galerkin method for Hamilton–Jacobi equations. *Appl. Numer. Math.* **33**, 423–434.
83. LeSaint, P., and Raviart, P. A. (1974). On a finite element method for solving the neutron transport equation. In de Boor, C. (ed.), *Mathematical Aspects of Finite Elements in Partial Differential Equations*, Academic Press, pp. 89–145.
84. LeVeque, R. J. (1990). *Numerical Methods for Conservation Laws*, Birkhäuser.
85. Lin, Q. (2000). Full convergence for hyperbolic finite elements. In Cockburn, B., Karniadakis, G. E., and Shu, C.-W. (eds.), *Discontinuous Galerkin Methods. Theory, Computation and Applications*, Lecture Notes in Computational Science and Engineering, Vol. 11, Springer-Verlag, pp. 167–177.
86. Lin, Q., Yan, N., and Zhou, A.-H. (1996). An optimal error estimate of the discontinuous Galerkin method. *J. Engrg. Math.* **13**, 101–105.
87. Lin, Q., and Zhou, A.-H. (1993). Convergence of the discontinuous Galerkin method for a scalar hyperbolic equation. *Acta Math. Sci.* **13**, 207–210.
88. Liu, J.-G., and Shu, C.-W. (2000). A high order discontinuous Galerkin method for 2D incompressible flows. *J. Comput. Phys.* **160**, 577–596.
89. Liu, J.-G., and Shu, C.-W. (2000). A numerical example on the performance of high-order discontinuous Galerkin method for 2D incompressible flows. In Cockburn, B., Karniadakis, G. E., and Shu, C.-W. (eds.), *Discontinuous Galerkin Methods. Theory, Computation and Applications*, Lecture Notes in Computational Science and Engineering, Vol. 11, Springer-Verlag, pp. 369–374.
90. Liu, J.-G., and Xin, Z.-P. (2000). Convergence of a Galerkin method for 2D discontinuous Euler flows. *Comm. Pure Appl. Math.* **53**, 786–798.
91. Lomtev, I., Kirby, R. M., and Karniadakis, G. E. (2000). A discontinuous Galerkin method in moving domains. In Cockburn, B., Karniadakis, G. E., and Shu, C.-W. (eds.), *Discontinuous Galerkin Methods. Theory, Computation and Applications*, Lecture Notes in Computational Science and Engineering, Vol. 11, Springer-Verlag, pp. 375–383.
92. Oden, J. T., Babuška, I., and Baumann, C. E. (1998). A discontinuous *hp* finite element method for diffusion problems. *J. Comput. Phys.* **146**, 491–519.
93. Osher, S. (1984). Convergence of generalized MUSCL schemes. *SIAM J. Numer. Anal.* **22**, 947–961.
94. Osher, S., and Sethian, J. A. (1988). Fronts propagating with curvature dependent speed: Algorithms based on Hamilton–Jacobi formulations. *J. Comput. Phys.* **79**, 12–49.
95. Osher, S., and Shu, C.-W. (1991). High-order essentially nonoscillatory schemes for Hamilton–Jacobi equations. *SIAM J. Numer. Anal.* **28**, 907–922.
96. Perugia, I. and Schötzau, D. The coupling of local discontinuous Galerkin and conforming finite element methods. *J. Sci. Comput.*, to appear.
97. Peterson, T. (1991). A note on the convergence of the discontinuous Galerkin method for a scalar hyperbolic equation. *SIAM J. Numer. Anal.* **28**, 133–140.

98. Reed, W. H., and Hill, T. R. *Triangular Mesh Methods for the Neutron Transport Equation*, Tech. Report LA-UR-73-479, Los Alamos Scientific Laboratory, 1973.
99. Richter, G. R. (1988). An optimal-order error estimate for the discontinuous Galerkin method. *Math. Comp.* **50**, 75–88.
100. Rivière, B., Wheeler, M. F., and Girault, V. (1999). Improved energy estimates for interior penalty, constrained and discontinuous Galerkin methods for elliptic problems. Part I. *Comput. Geom.* **3**, 337–360.
101. Rouy, E., and Tourin, A. (1992). A viscosity solutions approach to shape-from-shading. *SIAM J. Numer. Anal.* **29**, 867–884.
102. Sanders, R. (1983). On convergence of monotone finite difference schemes with variable spacing differencing. *Math. Comp.* **40**, 91–106.
103. Schwab, C. (1999). *hp*-FEM for fluid flow simulation. In Barth, T., and Deconink, H. (eds.), *High-Order Methods for Computational Physics*, Lecture Notes in Computational Science and Engineering, Vol. 9, Springer-Verlag, pp. 325–438.
104. Shu, C.-W. (1987). TVB boundary treatment for numerical solutions of conservation laws. *Math. Comp.* **49**, 123–134.
105. Shu, C.-W. (1987). TVB uniformly high order schemes for conservation laws. *Math. Comp.* **49**, 105–121.
106. Shu, C.-W. (1987). TVD time discretizations. *SIAM J. Sci. Stat. Comput.* **9** (1988), 1073–1084.
107. Shu, C.-W. (2001). Different formulations of the discontinuous Galerkin method for the viscous terms. In Shi, Z.-C., Mu, M., Xue, W., and Zou, J. (eds.), *Advances in Scientific Computing*, Science Press, pp. 144–155.
108. Shu, C.-W., and Osher, S. (1988). Efficient implementation of essentially non-oscillatory shock-capturing schemes. *J. Comput. Phys.* **77**, 439–471.
109. Shu, C.-W., and Osher, S. (1989). Efficient implementation of essentially non-oscillatory shock capturing schemes, II. *J. Comput. Phys.* **83**, 32–78.
110. Strang, G., and Fix, G. (1973). *An Analysis of the Finite Element Method*, Prentice-Hall, New Jersey.
111. van Leer, B. (1974). Towards the ultimate conservation difference scheme, II. *J. Comput. Phys.* **14**, 361–376.
112. van Leer, B. (1979). Towards the ultimate conservation difference scheme, V. *J. Comput. Phys.* **32**, 1–136.
113. Warburton, T. C. (1998). *Spectral/hp Methods on Polymorphic Multi-Domains: Algorithms and Applications*, Ph.D. thesis, Brown University.
114. Wheeler, M. F. (1978). An elliptic collocation-finite element method with interior penalties. *SIAM J. Numer. Anal.* **15**, 152–161.
115. Wierse, M. (1997). A new theoretically motivated higher order upwind scheme on unstructured grids of simplices. *Adv. Comput. Math.* **7**, 303–335.
116. Woodward, P., and Colella, P. (1984). The numerical simulation of two-dimensional fluid flow with strong shocks. *J. Comput. Phys.* **54**, 115–173.