AN INTRODUCTION TO THE DISCONTINUOUS GALERKIN METHOD FOR CONVECTION-DOMINATED PROBLEMS

Bernardo Cockburn

School of Mathematics, University of Minnesota,
Minneapolis, Minnesota 55455, USA
e-mail: cockburn@math.umn.edu

ABSTRACT

In these notes, we study the Runge Kutta Discontinuous Galerkin method for numericaly solving nonlinear hyperbolic systems and its extension for convection-dominated problems, the so-called Local Discontinuous Galerkin method. Examples of problems to which these methods can be applied are the Euler equations of gas dynamics, the shallow water equations, the equations of magneto-hydrodynamics, the compressible Navier-Stokes equations with high Reynolds numbers, and the equations of the hydrodynamic model for semiconductor device simulation. The main features that make the methods under consideration attractive are their formal high-order accuracy, their nonlinear stability, their high parallelizability, their ability to handle complicated geometries, and their ability to capture the discontinuities or strong gradients of the exact solution without producing spurious oscillations. The purpose of these notes is to provide a short introduction to the devising and analysis of these discontinuous Galerkin methods.

Aknowledgements. The author is grateful to Alfio Quarteroni for the invitation to give a series of lectures at the CIME, June 23–28, 1997, the material of which is contained in these notes. He also thanks F. Bassi and F. Rebay, and I. Lomtev and G.E. Karniadakis for kindly providing pictures from their papers [**2**] and [**3**], and [**46**] and [**65**], respectively.

Contents

Preface

There are several numerical methods using a DG formulation to discretize the equations in time, space, or both. In this monograph, we consider numerical methods that use DG discretizations *in space* and combine it with an *explicit* Runge-Kutta time-marching algorithm. We thus consider the so-called Runge-Kutta discontinuous Galerkin (RKDG) introduced and developed by Cockburn and Shu [**17, 15, 14, 13, 19**] for *nonlinear* hyperbolic systems and the so-called local discontinuous Galerkin (LDG) for *nonlinear* convection-diffusion systems. The LDG methods are an extension of the RKDG methods to convection-diffusion problems proposed first by Bassi and Rebay [**3**] in the context of the compressible Navier-Stokes and recently extended to general convection-diffusion problems by Cockburn and Shu [**18**].

Several properties are responsible for the increasing popularity of the above mentioned methods. The use of a DG discretization *in space* gives the methods the high-order accuracy, the flexibility in handling complicated geometries, and the easy to treat boundary conditions typical of the finite element methods. Moreover, the use of *discontinuous* elements produces a block-diagonal *mass* matrix whose blocks can be easily inverted by hand. This why after discretizing in time with a high-order accurate, *explicit* Runge-Kutta method, the resulting algorithm is highly parallelizable. Finally, these methods incorporate in a very natural way the techniques of 'slope limiting' developed by van Leer [**62, 63**] that effectively damp out the spurious oscillations that tend to be produced around the discontinuities or strong gradients of the approximate solution.

In these notes, we sudy these DG methods by following their historical development. Thus, we first study the RKDG method and then the LDG method. To study the RKDG method, we start by considering their definition for the scalar equation in one-space dimension. Then, we consider the scalar equation in several space dimensions and finally, we consider the case of multidimensional systems. The last chapter is devoted to the LDG methods.

To study the RKDG method, we take the point of view that they are formally high-order accurate 'perturbations' of the so-called 'monotone' schemes which are very stable and formally first-order accurate. Indeed, the RKDG methods were devised by trying to see if formally high-order accurate methods could be obtained that retained the remarkable stability of the monotone schemes. Of course, this approach is not new: It has been the basic idea in the devising of the so-called 'high-resolution' schemes for finite-difference and finite-volume methods for nonlinear conservation laws. Thus, the RKDG method incorporates this very successful idea into the framework of DG methods which have all the advantages of finite element methods.

CHAPTER 1

A historical overview

1.1. The original Discontinuous Galerkin method

The original discontinuous Galerkin (DG) finite element method was introduced by Reed and Hill [**54**] for solving the neutron transport equation

$$\sigma\, u + div(\overline{a}\, u) = f,$$

where σ is a real number and $\overline{a}$ a constant vector. Because of the linear nature of the equation, the approximate solution given by the method of Reed and Hill can be computed element by element when the elements are suitably ordered according to the characteristic direction.

LeSaint and Raviart [**41**] made the first analysis of this method and proved a rate of convergence of $(\Delta x)^k$ for general triangulations and of $(\Delta x)^{k+1}$ for Cartesian grids. Later, Johnson and Pitkäranta [**37**] proved a rate of convergence of $(\Delta x)^{k+1/2}$ for general triangulations and Peterson [**53**] confirmed this rate to be optimal. Richter [**55**] obtained the optimal rate of convergence of $(\Delta x)^{k+1}$ for some structured two-dimensional non-Cartesian grids.

1.2. Nonlinear hyperbolic systems: The RKDG method

The success of this method for linear equations, prompted several authors to try to extend the method to nonlinear hyperbolic conservation laws

$$u_t + \sum_{i=1}^{d} (f_i(u))_{x_i} = 0,$$

equipped with suitable initial or initial–boundary conditions. However, the introduction of the nonlinearity prevents the element-by-element computation of the solution. The scheme defines a nonlinear system of equations that must be solved all at once and this renders it computationally very inefficient for hyperbolic problems.

• **The one-dimensional scalar conservation law.**

To avoid this difficulty, Chavent and Salzano [**8**] contructed an explicit version of the DG method in the one-dimensional scalar conservation law. To do that, they discretized in space by using the DG method with piecewise linear elements and then discretized in time by using the simple Euler forward method. Although the resulting scheme is explicit, the classical von Neumann analysis shows that it is unconditionally unstable when the ratio $\frac{\Delta t}{\Delta x}$ is held constant; it is stable if $\frac{\Delta t}{\Delta x}$ is of order $\sqrt{\Delta x}$, which is a very restrictive condition for hyperbolic problems.

To improve the stability of the scheme, Chavent and Cockburn [**7**] modified the scheme by introducing a suitably defined 'slope limiter' following the ideas introduced by vanLeer in [**62**]. They thus obtained a scheme that was proven to be total variation diminishing in the means (TVDM) and total variation bounded (TVB) under a fixed CFL number, $f' \frac{\Delta t}{\Delta x}$, that can be chosen to be less than or equal to $1/2$. Convergence of a subsequence is thus guaranteed, and the numerical results given in [**7**] indicate convergence to the correct entropy solutions. On the other hand, the scheme is only first order accurate in time and the 'slope limiter' has to balance the spurious oscillations in smooth regions caused by linear instability, hence adversely affecting the quality of the approximation in these regions.

These difficulties were overcome by Cockburn and Shu in [**17**], where the first Runge Kutta Discontinuous Galerkin (RKDG) method was introduced. This method was contructed by (i) retaining the piecewise linear DG method for the space discretization, (ii) using a special explicit TVD second order Runge-Kutta type discretization introduced by Shu and Osher in a finite difference framework [**57**], [**58**], and (iii) modifying the 'slope limiter' to maintain the formal accuracy of the scheme at extrema. The resulting explicit scheme was then proven linearly stable for CFL numbers less than $1/3$, formally uniformly second order accurate in space and time including at extrema, and TVBM. Numerical results in [**17**] indicate good convergence behavior: Second order in smooth regions including at extrema, sharp shock transitions (usually in one or two elements) without oscillations, and convergence to entropy solutions even for non convex fluxes.

In [**15**], Cockburn and Shu extended this approach to construct (formally) high-order accurate RKDG methods for the scalar conservation law. To device RKDG methods of order $k+1$, they used (i) the DG method with polynomials of degree k for the space discretization, (ii) a TVD $(k+1)$-th order accurate explicit time discretization, and (iii) a generalized 'slope limiter.' The generalized 'slope limiter' was carefully devised with the purpose of enforcing the TVDM property without destroying the accuracy of the scheme. The numerical results in [**15**], for $k = 1, 2$, indicate $(k+1)$-th order order in smooth regions away from discontinuities as well as sharp shock transitions with no oscillations; convergence to the entropy solutions was observed in all the tests. These RKDG schemes were extended to one-dimensional systems in [**14**].

- **The multidimensional case.**

The extension of the RKDG method to the multidimensional case was done in [**13**] for the scalar conservation law. In the multidimensional case, the complicated geometry the spatial domain might have in practical applications can be easily handled by the DG space discretization. The TVD time discretizations remain the same, of course. Only the construction of the generalized 'slope limiter' represents a serious challenge. This is so, not only because of the more complicated form of the elements but also because of inherent accuracy barries imposed by the stability properties.

Indeed, since the main purpose of the 'slope limiter' is to enforce the nonlinear stability of the scheme, it is essential to realize that in the multidimensional case, the constraints imposed by the stability of a scheme on its accuracy are even greater than in the one dimensional case. Although in the one dimensional case it is possible to devise high-order accurate schemes with the TVD property, this is not true in several space dimensions since Goodman and LeVeque [**28**] proved that any TVD scheme is at most first order accurate. Thus, any generalized 'slope limiter' that enforces the TVD property, or the TVDM property for that matter, would unavoidably reduce the accuracy of the scheme to first-order accuracy. This is why in [**13**], Cockburn, Hou and Shu devised a generalized 'slope limiter' that enforced a **local** maximum principles only since they are not incompatible with high-order accuracy. No other class of schemes has a proven maximum principle for genearal nonlinearities $\mathbf{f}$, and arbitrary triangulations.

The extension of the RKDG methods to general multidimensional systems was started by Cockburn and Shu in [**16**] and has been recently completed in [**19**]. Bey and Oden [**5**] and more recently Bassi and Rebay [**2**] have studied applications of the method to the Euler equations of gas dynamics.

• **The main advantages of the RKDG method.**

The resulting RKDG schemes have several important advantages. First, like finite element methods such as the SUPG-method of Hughes and Brook [**29, 34, 30, 31, 32, 33**] (which has been analyzed by Johnson *et al* in [**38, 39, 40**]), the RKDG methods are better suited than finite difference methods to handle complicated geometries. Moreover, the particular finite elements of the DG space discretization allow an extremely simple treatment of the boundary conditions; no special numerical treatment of them is required in order to achieve uniform high order accuracy, as is the case for the finite difference schemes.

Second, the method can easily handle adaptivity strategies since the refining or unrefining of the grid can be done without taking into account the continuity restrictions typical of conforming finite element methods. Also, the degree of the approximating polynomial can be easily changed from one element to the other. Adaptivity is of particular importance in hyperbolic problems given the complexity of the structure of the discontinuities. In the one dimensional case the Riemann problem can be solved in closed form and discontinuity curves in the (x, t) plane are simple straight lines passing through the origin. However, in two dimensions their solutions display a very rich structure; see the works of Wagner [**64**], Lindquist [**43**], [**42**], Zhang and Zheng [**68**], and Zhang and Cheng [**67**]. Thus, methods which allow triangulations that can be easily adapted to resolve this structure, have an important advantage.

Third, the method is highly parallelizable. Since the elements are discontinuous, the mass matrix is block diagonal and since the order of the blocks is equal to the number of degrees of freedom inside the corresponding elements, the blocks can be inverted by hand once and for all. Thus, at each Runge-Kutta inner step, to update the degrees of freedom inside a given element, only the degrees of freedom of the elements sharing a face are involved; communication between processors is thus kept to a minimum. Extensive studies of adaptivity and parallelizability issues of the RKDG method were started by Biswas, Devine, and Flaherty [**6**] and then continued by deCougny *et al.* [**20**], Devine *et al.* [**22, 21**] and by Özturan *et al.* [**52**].

1.3. Convection-diffusion systems: The LDG method

The first extensions of the RKDG method to nonlinear, convection-diffusion systems of the form

$$\partial_t \mathbf{u} + \nabla \cdot \mathbf{F}(\mathbf{u}, D\,\mathbf{u}) = 0, \text{ in } (0,T) \times \Omega,$$

were proposed by Chen *et al.* [**10**], [**9**] in the framework of hydrodynamic models for semiconductor device simulation. In these extensions, approximations of second and third-order derivatives of the discontinuous approximate solution were obtained by using simple projections into suitable finite elements spaces. This projection requires the inversion of global mass matrices, which in [**10**] and [**9**] are 'lumped' in order to maintain the high parallelizability of the method. Since in [**10**] and [**9**] polynomials of degree one are used, the 'mass lumping' is justified; however, if polynomials of higher degree were used, the 'mass lumping' needed to enforce the full parallelizability of the method could cause a degradation of the formal order of accuracy.

Fortunately, this is not an issue with the methods proposed by Bassi and Rebay [**3**] (see also Bassi *et al* [**2**]) for the compressible Navier-Stokes equations. In these methods, the original idea of the RKDG method is applied to *both* u *and* $D\,u$ which are now considered as *independent* unknowns. Like the RKDG methods, the resulting methods are highly parallelizable methods of high-order accuracy which are very efficient for time-dependent, convection-dominated flows. The LDG methods considered by Cockburn and Shu [**18**] are a generalization of these methods.

The basic idea to construct the LDG methods is to *suitably rewrite* the original system as a larger, degenerate, first-order system and then discretize it by the RKDG method. By a careful choice of this rewriting, nonlinear stability can be achieved even without slope limiters, just as the RKDG method in the purely hyperbolic case; see Jiang and Shu [**36**].

The LDG methods [**18**] are very different from the so-called Discontinuous Galerkin (DG) method for parabolic problems introduced by Jamet [**35**] and studied by Eriksson, Johnson, and Thomée [**27**], Eriksson and Johnson [**23, 24, 25, 26**], and more recently by Makridakis and Babuška [**50**]. In the DG method, the approximate solution is discontinuous only in time, not in space; in fact, the space discretization is the standard Galerkin discretization with *continuous* finite elements. This is in strong contrast with the space discretizations of the LDG methods which use *discontinuous* finite elements. To emphasize this difference, those methods are called **Local** Discontinuous Galerkin methods. The large amount of degrees of freedom and the restrictive conditions of the size of the time step for explicit time-discretizations, render the LDG methods inefficient for diffusion-dominated problems; in this situation, the use of methods with continuous-in-space approximate solutions is recommended. However, as for the successful RKDG methods for purely hyperbolic problems, the extremely local domain of dependency of the LDG methods allows a very efficient parallelization that by far compensates for the extra amount of degrees of freedom in the case of convection-dominated flows.

Karniadakis *et al.* have implemented and tested these methods for the compressible Navier Stokes equations in two and three space dimensions with impressive results; see [**44**], [**45**], [**46**], [**47**], and [**65**].

1.4. The content of these notes

In these notes, we study the RKDG and LDG methods. Our exposition will be based on the papers by Cockburn and Shu [**17**], [**15**], [**14**], [**13**], and [**19**] in which the RKDG method was developed and on the paper by Cockburn and Shu [**18**] which is devoted to the LDG methods. Numerical results from the papers by Bassi and Rebay [**2**], on the Euler equations of gas dynamics, and [**3**], on the compressible Navier-Stokes equations, are also included.

The emphasis in these notes is on *how the above mentioned schemes were devised.* As a consequence, the chapters that follow reflect that development. Thus, Chapter 2, in which the RKDG schemes for the one-dimensional scalar conservation law are constructed, constitutes the core of the notes because it contains all the important ideas for the devicing of the RKDG methods; chapter 3 contains the extension to multidimensional systems; and chapter 4, the extension to convection-diffusion problems.

We would like to emphasize that the guiding principle in the devicing of the RKDG methods for scalar conservation laws is to consider them as *perturbations of the so-called monotone schemes.* As it is well-known, monotone schemes for scalar conservation laws are stable and converge to the entropy solution but are only first-order accurate. Following a widespread approach in the field of numerical schemes for nonlinear conservation laws, the RKDG are constructed in such a way that they are high-order accurate schemes that 'become' a monotone scheme when a piecewise-constant approximation is used. Thus, to obtain high-order accurate RKDG schemes, we 'perturb' the piecewise-constant approximation and allow it to be piecewise a polynomial of arbitrary degree. Then, the conditions under which the stability properties of the monotone schemes are still valid are sought and enforced by means of the generalized 'slope limiter.' The fact that it is possible to do so without destroying the accuracy of the RKDG method is the crucial point that makes this method both robust and accurate.

The issues of parallelization and adaptivity developed by Biswas, Devine, and Flaherty [**6**], deCougny *et al.* [**20**], Devine *et al.* [**22, 21**] and by Özturan *et al.* [**52**] are certainly very important. Another issue of importance is how to render the method computationaly more efficient, like the quadrature rule-free versions of the RKDG method recently studied by Atkins and Shu [**1**]. However, these topics fall beyond the scope of these notes whose main intention is to provide a simple introduction to the topic of discontinuous Galerkin methods for convection-dominated problems.

CHAPTER 2

The scalar conservation law in one space dimension

2.1. Introduction

In this section, we introduce and study the RKDG method for the following simple model problem:

$$u_t + f(u)_x = 0, \qquad \text{in } (0,1) \times (0,T), \tag{2.1.1}$$

$$u(x,0) = u_0(x), \qquad \forall\, x \in (0,1), \tag{2.1.2}$$

and periodic boundary conditions. This section has material drawn from [**17**] and [**15**].

2.2. The discontinuous Galerkin-space discretization

2.2.1. The weak formulation. To discretize in space, we proceed as follows. For each partition of the interval $(0,1)$, $\{\,x_{j+1/2}\,\}_{j=0}^{N}$, we set $I_j = (x_{j-1/2}, x_{j+1/2})$, $\Delta_j = x_{j+1/2} - x_{j-1/2}$ for $j = 1, \ldots, N$, and denote the quantity $\max_{1\le j\le N}\,\Delta_j$ by Δx .

We seek an approximation u_h to u such that for each time $t \in [0,T]$, $u_h(t)$ belongs to the finite dimensional space

$$V_h = V_h^k \equiv \{v \in L^1(0,1) : v|_{I_j} \in P^k(I_j),\ j = 1, \ldots, N\}, \tag{2.2.3}$$

where $P^k(I)$ denotes the space of polynomials in I of degree at most k. In order to determine the approximate solution u_h, we use a weak formulation that we obtain as follows. First, we multiply the equations (2.1.1) and (2.1.2) by arbitrary, smooth functions v and integrate over I_j, and get, after a simple formal integration by parts,

$$\begin{aligned}&\int_{I_j} \partial_t\, u(x,t)\, v(x)\, dx - \int_{I_j} f(u(x,t))\, \partial_x\, v(x)\, dx \\ &+ f(u(x_{j+1/2},t))\, v(x_{j+1/2}^-) - f(u(x_{j-1/2},t))\, v(x_{j-1/2}^+) = 0,\end{aligned} \tag{2.2.4}$$

$$\int_{I_j} u(x,0)\, v(x)\, dx = \int_{I_j} u_0(x)\, v(x)\, dx. \tag{2.2.5}$$

Next, we replace the smooth functions v by test functions v_h belonging to the finite element space V_h, and the exact solution u by the approximate solution u_h. Since the function u_h is discontinuous at the points $x_{j+1/2}$, we must also replace the nonlinear 'flux' $f(u(x_{j+1/2},t))$ by a *numerical* 'flux' that depends on the two values of u_h at the point $(x_{j+1/2},t)$, that is, by the function

$$h(u)_{j+1/2}(t) = h(u(x^-_{j+1/2},t), u(x^+_{j+1/2},t)), \tag{2.2.6}$$

that will be suitably chosen later. Note that *we always use the same numerical flux regardless of the form of the finite element space.* Thus, the approximate solution given by the DG-space discretization is defined as the solution of the following weak formulation:

$$\forall\ j=1,\ldots,N, \qquad \forall\ v_h \in P^k(I_j):$$

$$\begin{aligned}
&\int_{I_j} \partial_t\, u_h(x,t)\, v_h(x)\, dx - \int_{I_j} f(u_h(x,t))\, \partial_x\, v_h(x)\, dx \\
&+h(u_h)_{j+1/2}(t)\, v_h(x^-_{j+1/2}) - h(u_h)_{j-1/2}(t)\, v_h(x^+_{j-1/2}) = 0,
\end{aligned} \tag{2.2.7}$$

$$\int_{I_j} u_h(x,0)\, v_h(x)\, dx = \int_{I_j} u_0(x)\, v_h(x)\, dx. \tag{2.2.8}$$

2.2.2. Incorporating the monotone numerical fluxes. To complete the definition of the approximate solution u_h, it only remains to choose the numerical flux h. To do that, we invoke our main point of view, namely, that *we want to construct schemes that are perturbations of the so-called monotone schemes* because monotone schemes, although only first-order accurate, are very stable and converge to the entropy solution. More precisely, we want that in the case $k=0$, that is, when the approximate solution u_h is a piecewise-constant function, our DG-space discretization gives rise to a monotone scheme.

Since in this case, for $x \in I_j$ we can write

$$u_h(x,t) = u^0_j,$$

we can rewrite our weak formulation (2.2.7), (2.2.8) as follows:

$$\forall\ j=1,\ldots,N:$$

$$\begin{aligned}
&\partial_t\, u^0_j(t) + \left\{h(u^0_j(t), u^0_{j+1}(t)) - h(u^0_{j-1}(t), u^0_j(t))\right\}/\Delta_j = 0,\\
&u^0_j(0) = \frac{1}{\Delta_j}\int_{I_j} u_0(x)\, dx,
\end{aligned}$$

and it is well-known that this defines a monotone scheme if $h(a,b)$ is a Lipschitz, consistent, monotone flux, that is, if it is,

(i) locally Lipschitz and consistent with the flux $f(u)$, i.e., $h(u,u) = f(u)$,
(ii) a nondecreasing function of its first argument, and
(iii) a nonincreasing function of its second argument.

The best-known examples of numerical fluxes satisfying the above properties are the following:

(i) The Godunov flux:

$$h^G(a,b) = \begin{cases} \min_{a \le u \le b} f(u) \ , & \text{if } a \le b, \\ \max_{a \ge u \ge b} f(u) \ , & \text{if } a > b; \end{cases}$$

(ii) The Engquist-Osher flux:

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

(iii) The Lax-Friedrichs flux:

$$\begin{aligned} h^{LF}(a,b) &= \frac{1}{2} \left[f(a) + f(b) - C\,(b-a) \right], \\ C &= \max_{\inf u^0(x) \le s \le \sup u^0(x)} |f'(s)|; \end{aligned}$$

(iv) The local Lax–Friedrichs flux:

$$\begin{aligned} h^{LLF}(a,b) &= \frac{1}{2} \left[f(a) + f(b) - C(b-a) \right], \\ C &= \max_{\min(a,b) \le s \le \max(a,b)} |f'(s)|; \end{aligned}$$

(v) The Roe flux with 'entropy fix':

$$h^R(a,b) = \begin{cases} f(a), & \text{if } f'(u) \ge 0 \quad \text{for} \quad u \in [\min(a,b),\ \max(a,b)], \\ f(b), & \text{if } f'(u) \le 0 \quad \text{for} \quad u \in [\min(a,b), \max(a,b)], \\ h^{LLF}(a,b), & \text{otherwise.} \end{cases}$$

For the flux h, we can use the Godunov flux h^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 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 Lax-Friedrichs flux. However, numerical experience suggests that as the degree k of the approximate solution increases, the choice of the numerical flux does not have a significant impact on the quality of the approximations.

2.2.3. Diagonalizing the mass matrix. If we choose the Legendre polynomials P_ℓ as local basis functions, we can exploit their L^2-orthogonality, namely,

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

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

$$u_h(x,t) \quad = \quad \sum_{\ell=0}^{k} u_j^\ell\, \varphi_\ell(x),$$

where

$$\varphi_\ell(x) = P_\ell(2\,(x - x_j)/\Delta_j),$$

the weak formulation (2.2.7), (2.2.8) takes the following simple form:

$$\begin{aligned}
&\forall\ j = 1, \dots, N \text{ and } \ell = 0, \dots, k :\\
&\left(\frac{1}{2\ell+1}\right) \partial_t\, u_j^\ell(t) - \frac{1}{\Delta_j} \int_{I_j} f(u_h(x,t))\, \partial_x \varphi_\ell(x)\, dx\\
&+ \frac{1}{\Delta_j} \Big\{ h(u_h(x_{j+1/2}))(t) - (-1)^\ell\, h(u_h(x_{j-1/2}))(t) \Big\} = 0,\\
&u_j^\ell(0) = \frac{2\ell+1}{\Delta_j} \int_{I_j} u_0(x)\, \varphi_\ell(x)\, dx,
\end{aligned}$$

where we have use the following properties of the Legendre polynomials:

$$P_\ell(1) = 1, \qquad P_\ell(-1) = (-1)^\ell.$$

This shows that after discretizing in space the problem (2.1.1), (2.1.2) by the DG method, we obtain a system of ODEs for the degrees of freedom that we can rewrite as follows:

$$\frac{d}{dt}\, u_h = L_h(u_h), \qquad \text{in } (0, T), \tag{2.2.9}$$

$$u_h(t = 0) = u_{0h}. \tag{2.2.10}$$

The element $L_h(u_h)$ of V_h is, of course, the approximation to $-f(u)_x$ provided by the DG-space discretization.

Note that if we choose a different local basis, the local mass matrix could be a full matrix but it will always be a matrix of order $(k+1)$. By inverting it by means of a symbolic manipulator, we can always write the equations for the degrees of freedom of u_h as an ODE system of the form above.

2.2.4. Convergence analysis of the linear case. In the linear case $f(u) = c\,u$, the $L^\infty(0, T; L^2(0, 1))$-accuracy of the method (2.2.7), (2.2.8) can be established by using the $L^\infty(0, T; L^2(0, 1))$-stability of the method and the approximation properties of the finite element space V_h.

Note that in this case, all the fluxes displayed in the examples above coincide and are equal to

$$h(a, b) = c\,\frac{a+b}{2} - \frac{|\,c\,|}{2}(b - a). \tag{2.2.11}$$

The following results are thus for this numerical flux.

We state the L^2-stability result in terms of the jumps of u_h across $x_{j+1/2}$ which we denote by

$$[u_h]_{j+1/2} \equiv u_h(x^+_{j+1/2}) - u_h(x^-_{j+1/2}).$$

PROPOSITION 2.1. (L^2-stability) We have,

$$\tfrac{1}{2}\| u_h(T) \|^2_{L^2(0,1)} + \Theta_T(u_h) \le \tfrac{1}{2}\| u_0 \|^2_{L^2(0,1)},$$

where

$$\Theta_T(u_h) = \tfrac{|c|}{2} \int_0^T \sum_{1\le j\le N} [u_h(t)]^2_{j+1/2}\, dt.$$

Note how the jumps of u_h are controled by the L^2-norm of the initial condition. This control reflects the subtle built-in dissipation mechanism of the DG-methods and is what allows the DG-methods to be more accurate than the standard Galerkin methods. Indeed, the standard Galerkin method has an order of accuracy equal to k whereas the DG-methods have an order of accuray equal to $k + 1/2$ for the same smoothness of the initial condition.

THEOREM 2.1. *Suppose that the initial condition u_0 belongs to $H^{k+1}(0,1)$. Let e be the approximation error $u - u_h$. Then we have,*

$$\| e(T) \|_{L^2(0,1)} \le C\, | u_0 |_{H^{k+1}(0,1)} (\Delta x)^{k+1/2},$$

where C depends solely on k, $|c|$, and T.

It is also possible to prove the following result if we assume that the initial condition is more regular. Indeed, we have the following result.

THEOREM 2.2. *Suppose that the initial condition u_0 belongs to $H^{k+2}(0,1)$. Let e be the approximation error $u - u_h$. Then we have,*

$$\| e(T) \|_{L^2(0,1)} \le C\, | u_0 |_{H^{k+2}(0,1)} (\Delta x)^{k+1},$$

where C depends solely on k, $|c|$, and T.

The Theorem 2.1 is a simplified version of a more general result proven in 1986 by Johnson and Pitkäranta [**37**] and the Theorem 2.2 is a simplified version of a more general result proven in 1974 by LeSaint and Raviart [**41**]. To provide a simple introduction to the techniques used in these more general results, we give *new* proofs of these theorems in an appendix to this chapter.

The above theorems show that the DG-space discretization results in a $(k+1)$th-order accurate scheme, at least in the linear case. This gives a strong indication that the same order of accuracy should hold in the nonlinear case when the exact solution is smooth enough, of course.

Now that we know that the DG-space discretization produces a high-order accurate scheme for smooth exact solutions, we consider the question of how does it behave when the flux is a nonlinear function.

2.2.5. Convergence analysis in the nonlinear case. To study the convergence properties of the DG-method, we first study the convergence properties of the solution w of the following problem:

$$w_t + f(w)_x = (\nu(w)\, w_x)_x, \qquad \text{in } (0,1)\times(0,T), \tag{2.2.12}$$

$$w(x,0) = u_0(x), \qquad \forall\, x \in (0,1), \tag{2.2.13}$$

and periodic boundary conditions. We then mimic the procedure to study the convergence of the DG-method for the piecewise-constant case. The general DG-method will be considered later after having introduced the Runge-Kutta time-discretization.

The continuous case as a model. In order to compare u and w, it is *enough* to have (i) an entropy inequality and (ii) uniform boundedness of $\| w_x \|_{L^1(0,1)}$. Next, we show how to obtain these properties in a formal way.

We start with the entropy inequality. To obtain such an inequality, the basic idea is to multiply the equation (2.2.12) by $U'(w-c)$, where $U(\cdot)$ denotes the absolute value function and c denotes an arbitrary real number. Since

$$\begin{aligned}
U'(w-c)\, w_t &= U(w-c)_t,\\
U'(w-c)\, f(w)_x &= \big(U'(w-c)\,(f(w)-f(c))\big) \equiv F(w,c)_x,\\
U'(w-c)\,(\nu(w)\, w_x)_x &= \left(\int_c^w U'(\rho-c)\,\nu(\rho)\,d\rho\right)_{xx} - U''(w-c)\,\nu(w)\,(w_x)^2\\
&\equiv \Phi(w,c)_{xx} - U''(w-c)\,\nu(w)\,(w_x)^2,
\end{aligned}$$

we obtain

$$U(w-c)_t + F(w,c)_x - \Phi(w,c)_{xx} \leq 0, \qquad \text{in } (0,1)\times(0,T),$$

which is nothing but the entropy inequality we wanted.

To obtain the uniform boundedness of $\| w_x \|_{L^1(0,1)}$, the idea is to multiply the equation (2.2.12) by $-(U'(w_x))_x$ and integrate on x from 0 to 1. Since

$$\begin{aligned}
\int_0^1 -(U'(w_x))_x\, w_t &= \int_0^1 U'(w_x)\,(w_x)_t = \frac{d}{dt}\| w_x \|_{L^1(0,1)},\\
\int_0^1 -(U'(w_x))_x\, f(w)_x &= -\int_0^1 U''(w_x)\, w_{xx}\, f'(w)\, w_x = 0,\\
\int_0^1 -(U'(w_x))_x\,(\nu(w)\, w_x)_x &= -\int_0^1 U''(w_x)\, w_{xx}\,(\nu'(w)\,(w_x)^2 + \nu(w)\, w_{xx})\\
&= -\int_0^1 U''(w_x)\,\nu(w)\,(w_{xx})^2 \leq 0,
\end{aligned}$$

we immediately get that

$$\frac{d}{dt}\| w_x \|_{L^1(0,1)} \leq 0,$$

and so,

$$\| w_x \|_{L^1(0,1)} \leq \| (u_0)_x \|_{L^1(0,1)}, \qquad \forall\, t \in (0,T).$$

When the function u_0 has discontinuities, the same result holds with the total variation of u_0 ,$| u_0 |_{TV(0,1)}$, replacing the quantity $\| (u_0)_x \|_{L^1(0,1)}$; these two quantities coincide when $u_0 \in W^{1,1}(0,1)$.

With the two above ingredients, the following error estimate, obtained in 1976 by Kuznetsov, can be proved:

THEOREM 2.3. *We have*

$$\| u(T) - w(T) \|_{L^1(0,1)} \quad \leq \quad | u_0 |_{TV(0,1)} \sqrt{8\, T\, \nu},$$

where $\nu = \sup_{s\in[\inf u_0, \sup u_0]} \nu(s)$.

The piecewise-constant case. Let consider the simple case of the DG-method that uses a piecewise-constant approximate solution:

$$\forall\ j = 1, \ldots, N:$$

$$\begin{aligned}
&\partial_t u_j + \big\{h(u_j, u_{j+1}) - h(u_{j-1}, u_j)\big\}/\Delta_j = 0,\\
&u_j(0) = \frac{1}{\Delta_j}\int_{I_j} u_0(x)\,dx,
\end{aligned}$$

where we have dropped the superindex '0.' We pick the numerical flux h to be the Engquist-Osher flux.

According to the model provided by the continuous case, we must obtain (i) an entropy inequality and (ii) the uniform boundedness of the total variation of u_h.

To obtain the entropy inequality, we multiply our equation by $U'(u_j - c)$:

$$\partial_t U(u_j - c) + U'(u_j - c)\big\{h(u_j, u_{j+1}) - h(u_{j-1}, u_j)\big\}/\Delta_j = 0.$$

The second term in the above equation needs to be carefully treated. First, we rewrite the Engquist-Osher flux in the following form:

$$h^{EO}(a, b) = f^+(a) + f^-(b),$$

and, accordingly, rewrite the second term of the equality above as follows:

$$ST_j \;=\; U'(u_j - c)\big\{f^+(u_j) - f^+(u_{j-1})\big\} + U'(u_j - c)\big\{f^-(u_{j+1}) - f^-(u_j)\big\}.$$

Using the simple identity

$$U'(a - c)(g(a) - g(b)) = G(a, c) - G(b, c) + \int_a^b (g(b) - g(\rho))\,U''(\rho - x)\,d\rho.$$

where $G(a, c) = \int_c^a U'(\rho - c)\,g(\rho)\,d\rho$, we get

$$\begin{aligned}
ST_j \;&=\; F^+(u_j, c) - F^+(u_{j-1}, c) + \int_{u_j}^{u_{j-1}} (f^+(u_{j-1}) - f^+(\rho))\,U''(\rho - x)\,d\rho\\
&\quad + F^-(u_{j+1}, c) - F^-(u_j, c) - \int_{u_j}^{u_{j+1}} (f^-(u_{j+1}) - f^-(\rho))\,U''(\rho - x)\,d\rho\\
&=\; F(u_j, u_{j+1}; c) - F(u_{j-1}, u_j; c) + \Theta_{diss,j}
\end{aligned}$$

where

$$\begin{aligned}
F(a, b; c) \;&=\; F^+(a, c) + F^-(b, c),\\
\Theta_{diss,j} \;&=\; +\int_{u_j}^{u_{j-1}} (f^+(u_{j-1}) - f^+(\rho))\,U''(\rho - x)\,d\rho\\
&\quad - \int_{u_j}^{u_{j+1}} (f^-(u_{j+1}) - f^-(\rho))\,U''(\rho - x)\,d\rho.
\end{aligned}$$

We thus get

$$\partial_t U(u_j - c) + \big\{F(u_j, u_{j+1}; c) - F(u_{j-1}, u_j; c)\big\}/\Delta_j + \Theta_{diss,j}/\Delta_j = 0.$$

Since, f^+ and $-f^-$ are nondecreasing functions, we easily see that

$$\Theta_{diss,j} \quad \geq \quad 0,$$

and we obtain our entropy inequality:

$$\partial_t U(u_j - c) + \big\{F(u_j, u_{j+1}; c) - F(u_{j-1}, u_j; c)\big\}/\Delta_j \leq 0.$$

Next, we obtain the uniform boundedness on the total variation. To do that, we follow our model and multiply our equation by a discrete version of $-(U'(w_x))_x$, namely,

$$v_j^0 \quad = \quad -\frac{1}{\Delta_j}\left\{U'\left(\frac{u_{j+1} - u_j}{\Delta_{j+1/2}}\right) - U'\left(\frac{u_j - u_{j-1}}{\Delta_{j-1/2}}\right)\right\},$$

where $\Delta_{j+1/2} = (\Delta_j + \Delta_{j+1})/2$, multiply it by Δ_j and sum over j from 1 to N. We easily obtain

$$\frac{d}{dt} |\, u_h\,|_{TV(0,1)} + \sum_{1\leq j\leq N} v_j^0\,\{h(u_j, u_{j+1}) - h(u_{j-1}, u_j)\} = 0,$$

where

$$|\, u_h\,|_{TV(0,1)} \equiv \sum_{1\leq j\leq N} |\, u_{j+1} - u_j\,|.$$

According to our continuous model, the second term in the above equality should be positive. Let us see that this is indeed the case:

$$\begin{aligned} v_j^0\,\{h(u_j, u_{j+1}) - h(u_{j-1}, u_j)\} \quad &= \quad v_j^0\,\{f^+(u_j) - f^+(u_{j-1})\} + v_j^0\,\{f^-(u_{j+1}) - f^-(u_j)\} \\ &\geq \quad 0, \end{aligned}$$

by the definition of v_j^0, f^+, and f^-. This implies that

$$|\, u_h(t)\,|_{TV(0,1)} \leq |\, u_h(0)\,|_{TV(0,1)} \leq |\, u_0\,|_{TV(0,1)}.$$

With the two above ingredients, the following error estimate, obtained in 1976 by Kuznetsov, can be proved:

Theorem 2.4. *We have*

$$\|\, u(T) - u_h(T)\,\|_{L^1(0,1)} \quad \leq \quad \|\, u_0 - u_h(0)\,\|_{L^1(0,1)} + C\,|\, u_0\,|_{TV(0,1)}\sqrt{T\,\Delta x}.$$

2.3. The TVD-Runge-Kutta time discretization

To discretize our ODE system in time, we use the TVD Runge Kutta time discretization introduced in [**60**]; see also [**57**] and [**58**].

2.3.1. The discretization. Thus, if $\{t^n\}_{n=0}^N$ is a partition of $[0,T]$ and $\Delta t^n = t^{n+1} - t^n, n = 0, ..., N-1$, our time-marching algorithm reads as follows:

- Set $u_h^0 = u_{0h}$;
- For $n = 0, ..., N-1$ compute u_h^{n+1} from u_h^n as follows:
  1. set $u_h^{(0)} = u_h^n$;
  2. for $i = 1, ..., k+1$ compute the intermediate functions:
  $$u_h^{(i)} = \left\{ \sum_{l=0}^{i-1} \alpha_{il} u_h^{(l)} + \beta_{il} \Delta t^n L_h(u_h^{(l)}) \right\};$$
  3. set $u_h^{n+1} = u_h^{(k+1)}$.

Note that this method is very easy to code since *only a single subroutine defining* $L_h(u_h)$ *is needed*. Some Runge-Kutta time discretization parameters are displayed on the table below.

Table 1

Parameters of some practical Runge-Kutta time discretizations			
order	α_{il}	β_{il}	$\max\{\beta_{il}/\alpha_{il}\}$
2	1 $\frac{1}{2}\ \frac{1}{2}$	1 $0\ \frac{1}{2}$	1
3	1 $\frac{3}{4}\ \frac{1}{4}$ $\frac{1}{3}\ 0\ \frac{2}{3}$	1 $0\ \frac{1}{4}$ $0\ 0\ \frac{2}{3}$	1

2.3.2. The stability property. Note that all the values of the parameters α_{il} displayed in the table below are nonnegative; this is not an accident. Indeed, this is a condition on the parameters α_{il} that ensures the stability property

$$|\, u_h^{n+1} \,| \leq |\, u_h^n \,|,$$

provided that the 'local' stability property

$$|\, w \,| \leq |\, v \,|, \tag{2.3.14}$$

where w is obtained from v by the following 'Euler forward' step,

$$w = v + \delta\ L_h(v), \tag{2.3.15}$$

holds for values of $|\, \delta \,|$ smaller than a given number δ_0.

For example, the second-order Runke-Kutta method displayed in the table above can be rewritten as follows:

$$u_h^{(1)} = u_h^n + \Delta t\, L_h(u_h^n),$$
$$w_h = u_h^{(1)} + \Delta t\, L_h(u_h^{(1)}),$$
$$u_h^{n+1} = \frac{1}{2}(u_h^n + w_h).$$

Now, assuming that the stability property (2.3.14), (2.3.15) is satisfied for

$$\delta_0 = |\,\Delta t\ \max\{\beta_{il}/\alpha_{il}\}\,| = \Delta t,$$

we have

$$|\,u_h^{(1)}\,| \le |\,u_h^n\,|, \qquad |\,w_h\,| \le |\,u_h^{(1)}\,|,$$

and so,

$$|\,u_h^{n+1}\,| \le \frac{1}{2}(\,|\,u_h^n\,| + |\,w_h\,|) \le \,|\,u_h^n\,|.$$

Note that we can obtain this result because the coefficients α_{il} are positive! Runge-Kutta methods of this type of order up to order 5 can be found in [**58**].

The above example shows how to prove the following more general result.

THEOREM 2.5. *Assume that the stability property for the single 'Euler forward' step (2.3.14), (2.3.15) is satisfied for*

$$\delta_0 = \max_{0\le n\le N} |\,\Delta t^n\ \max\{\beta_{il}/\alpha_{il}\}\,|.$$

Assume also that all the coeficients α_{il} are nonnegative and satisfy the following condition:

$$\sum_{l=0}^{i-1} \alpha_{il} = 1, \qquad i = 1, \ldots, k+1.$$

Then

$$|\,u_h^n\,| \le |\,u_h^0\,|, \qquad \forall\, n \ge 0.$$

This stability property of the TVD-Runge-Kutta methods is crucial since it allows us to obtain the stability of the method from the stability of a single 'Euler forward' step.

Proof of Theorem 2.5. We start by rewriting our time discretization as follows:

- Set $u_h^0 = u_{0h}$;
- For $n = 0, ..., N-1$ compute u_h^{n+1} from u_h^n as follows:
  1. set $u_h^{(0)} = u_h^n$;
  2. for $i = 1, ..., k+1$ compute the intermediate functions:
  $$u_h^{(i)} = \sum_{l=0}^{i-1} \alpha_{il}\, w_h^{(il)},$$
  where
  $$w_h^{(il)} = u_h^{(l)} + \frac{\beta_{il}}{\alpha_{il}}\, \Delta t^n\, L_h(u_h^{(l)});$$
  3. set $u_h^{n+1} = u_h^{(k+1)}$.

We then have

$$
\begin{aligned}
|\,u_h^{(i)}\,| &\le \sum_{l=0}^{i-1} \alpha_{il}\,|\,w_h^{(il)}\,|, \quad \text{since } \alpha_{il} \ge 0, \\
&\le \sum_{l=0}^{i-1} \alpha_{il}\,|\,u_h^{(l)}\,|, \quad \text{by the stability property (2.3.14), (2.3.15),} \\
&\le \max_{0\le l\le i-1} |\,u_h^{(l)}\,|, \quad \text{since } \sum_{l=0}^{i-1} \alpha_{il} = 1.
\end{aligned}
$$

It is clear now that that Theorem 2.5 follows from the above inequality by a simple induction argument. $\square$

2.3.3. Remarks about the stability in the linear case. For the linear case $f(u) = c\,u$, Chavent and Cockburn [**7**] proved that for the case $k = 1$, i.e., for piecewise-linear approximate solutions, the single 'Euler forward' step is *unconditionally* $L^\infty(0,T;L^2(0,1))$-unstable for any fixed ratio $\Delta t/\Delta x$. On the other hand, in [**17**] it was shown that if a Runge-Kutta method of second order is used, the scheme is $L^\infty(0,T;L^2(0,1))$-stable provided that

$$
c\,\frac{\Delta t}{\Delta x} \le \frac{1}{3}.
$$

This means that we cannot deduce the stability of the complete Runge-Kutta method from the stability of the single 'Euler forward' step. As a consequence, we cannot apply Theorem 2.5 and we must consider the complete method at once.

Our numerical experiments show that when polynomial of degree k are used, a Runge-Kutta of order $(k+1)$ must be used. In this case, the $L^\infty(0,T;L^2(0,1))$-stability condition is the following:

$$
c\,\frac{\Delta t}{\Delta x} \le \frac{1}{2k+1}.
$$

There is no rigorous proof of this fact yet.

At a first glance, this stability condition, also called the Courant-Friedrichs-Levy (CFL) condition, seems to compare unfavorably with that of the well-known finite difference schemes. However, we must remember that in the DG-methods there are $(k+1)$ degrees of freedom in each element of size Δx whereas for finite difference schemes there is a single degree of freedom of each cell of size Δx. Also, if a finite difference scheme is of order $(k+1)$ its so-called stencil must be of at least $(2k+1)$ points, whereas the DG-scheme has a stencil of $(k+1)$ elements only.

2.3.4. Convergence analysis in the nonlinear case. Now, we explore what is the impact of the explicit Runge-Kutta time-discretization on the convergence properties of the methods under consideration. We start by considering the piecewise-constant case.

The piecewise-constant case. Let us begin by considering the simplest case, namely,

$$\forall\ j = 1, \ldots, N:$$

$$\begin{aligned}&(\,u_j^{n+1} - u_j^n\,)/\Delta t + \big\{h(u_j^n, u_{j+1}^n) - h(u_{j-1}^n, u_j^n)\big\}/\Delta_j = 0,\\ &u_j(0) = \frac{1}{\Delta_j}\int_{I_j} u_0(x)\,dx,\end{aligned}$$

where we pick the numerical flux h to be the Engquist-Osher flux.

According to the model provided by the continuous case, we must obtain (i) an entropy inequality and (ii) the uniform boundedness of the total variation of u_h.

To obtain the entropy inequality, we proceed as in the semidiscrete case and obtain the following result; see [**12**] for details.

THEOREM 2.6. *We have*

$$\begin{aligned}\big\{U(u_j^{n+1} - c) - U(u_j^n - c)\big\}/\Delta t \ &+ \ \big\{F(u_j^n, u_{j+1}^n; c) - F(u_{j-1}^n, u_j^n; c)\big\}/\Delta_j\\ &+ \ \Theta_{diss,j}^n/\Delta t = 0,\end{aligned}$$

where

$$\begin{aligned}\Theta_{diss,j}^n \ &= \ \int_{u_j^{n+1}}^{u_j^n} (p_j(u_j^n) - p_j(\rho))\,U''(\rho - x)\,d\rho\\ &\quad + \frac{\Delta t}{\Delta_j}\int_{u_j^{n+1}}^{u_{j-1}^n} (f^+(u_{j-1}^n) - f^+(\rho))\,U''(\rho - x)\,d\rho\\ &\quad - \frac{\Delta t}{\Delta_j}\int_{u_j^{n+1}}^{u_{j+1}^n} (f^-(u_{j+1}^n) - f^-(\rho))\,U''(\rho - x)\,d\rho,\end{aligned}$$

and

$$p_j(w) \ = \ w - \frac{\Delta t}{\Delta_j}(f^+(w) - f^-(w)).$$

Moreover, if the following CFL condition is satisfied

$$\max_{1\le j\le N} \frac{\Delta t}{\Delta_j}\,|\,f'\,| \ \ \le \ \ 1,$$

then $\Theta_{diss,j}^n \ge 0$, and the following entropy inequality holds:

$$\big\{U(u_j^{n+1} - c) - U(u_j^n - c)\big\}/\Delta t + \big\{F(u_j^n, u_{j+1}; c) - F(u_{j-1}, u_j; c)\big\}/\Delta_j \le 0.$$

Note that $\Theta_{diss,j}^n \ge 0$ because f^+, $-f^-$, are nondecreasing and because p_j is also nondecreasing under the above CFL condition.

Next, we obtain the uniform boundedness on the total variation. Proceding as before, we easily obtain the following result.

THEOREM 2.7. *We have*

$$|\,u_h^{n+1}\,|_{TV(0,1)} - |\,u_h^n\,|_{TV(0,1)} + \Theta_{TV}^n \ \ = \ \ 0,$$

where

$$\begin{aligned}
\Theta^n_{TV} &= \sum_{1\le j\le N} \left({U'}^{n}_{j+1/2} - {U'}^{n+1}_{j+1/2} \right) (p_{j+1/2}(u^n_{j+1}) - p_{j+1/2}(u^n_j) \\
&\quad + \sum_{1\le j\le N} \frac{\Delta t}{\Delta_j} \left({U'}^{n}_{j-1/2} - {U'}^{n+1}_{j+1/2} \right) (f^+(u^n_j) - f^+(u^n_{j-1})) \\
&\quad - \sum_{1\le j\le N} \frac{\Delta t}{\Delta_j} \left({U'}^{n}_{j+1/2} - {U'}^{n+1}_{j-1/2} \right) (f^-(u^n_{j+1}) - f^-(u^n_j))
\end{aligned}$$

where

$${U'}^{m}_{i+1/2} = U'\left(\frac{u^m_{i+1} - u^m_i}{\Delta_{i+1/2}} \right),$$

and

$$p_{j+1/2}(w) \quad = \quad s - \frac{\Delta t}{\Delta_{j+1}}\, f^+(w) + \frac{\Delta t}{\Delta_j}\, f^-(w).$$

Moreover, if the following CFL condition is satisfied

$$\max_{1\le j\le N} \frac{\Delta t}{\Delta_j} \,|\, f' \,| \quad \le \quad 1,$$

then $\Theta^n_{TV} \ge 0$*, and we have*

$$|\, u^n_h \,|_{TV(0,1)} \quad \le \quad |\, u_0 \,|_{TV(0,1)}.$$

With the two above ingredients, the following error estimate, obtained in 1976 by Kuznetsov, can be proved:

THEOREM 2.8. *We have*

$$\|\, u(T) - u_h(T) \,\|_{L^1(0,1)} \quad \le \quad \|\, u_0 - u_h(0) \,\|_{L^1(0,1)} + C \,|\, u_0 \,|_{TV(0,1)} \sqrt{T\, \Delta x}.$$

The general case. The study of the general case is much more difficult than the study of the monotone schemes. In these notes, we restrict ourselves to the study of the stability of the RKDG schemes. Hence, we restrict ourselves to the task of studying under what conditions the total variation of the *local means* is uniformly bounded.

If we denote by $\overline{u}_j$ the mean of u_h on the interval I_j, by setting $v_h = 1$ in the equation (2.2.7), we obtain,

$$\forall\ j = 1, \ldots, N:$$

$$(\overline{u}_j)_t + \left\{ h(u^-_{j+1/2}, u^+_{j+1/2}) - h(u^-_{j-1/2}, u^+_{j-1/2}) \right\} / \Delta_j = 0,$$

where $u^-_{j+1/2}$ denotes the limit from the left and $u^+_{j+1/2}$ the limit from the right. We pick the numerical flux h to be the Engquist-Osher flux.

This shows that if we set w_h equal to the Euler forward step $u_h + \delta\, L_h(u_h)$, we obtain

$$\forall\ j = 1, \ldots, N :$$

$$(\overline{w}_j - \overline{u}_j\,)/\delta + \left\{h(u^-_{j+1/2}, u^+_{j+1/2}) - h(u^-_{j-1/2}, u^+_{j-1/2})\right\}/\Delta_j = 0.$$

Proceeding exactly as in the piecewise-constant case, we obtain the following result for the total variation of the avergages,

$$|\,\overline{u}_h\,|_{TV(0,1)} \quad\equiv \sum_{1\le j\le N} |\,\overline{u}_{j+1} - \overline{u}_j\,|.$$

THEOREM 2.9. *We have*

$$|\,\overline{w}_h\,|_{TV(0,1)} - |\,\overline{u}_h\,|_{TV(0,1)} + \Theta_{TVM} \quad = \quad 0,$$

where

$$\begin{aligned}\Theta_{TVM} \quad = \quad & \sum_{1\le j\le N} \left(U'_{j+1/2} - U'_{j+1/2}\right) (p_{j+1/2}(u_h|_{I_{j+1}}) - p_{j+1/2}(u_h|_{I_j})\\ & + \sum_{1\le j\le N} \frac{\delta}{\Delta_j}\left(U'_{j-1/2} - U'_{j+1/2}\right)(f^+(u^-_{j+1/2}) - f^+(u^-_{j-1/2}))\\ & - \sum_{1\le j\le N} \frac{\delta}{\Delta_j}\left(U'_{j+1/2} - U'_{j-1/2}\right)(f^-(u^+_{j+1/2}) - f^-(u^+_{j-1/2}))\end{aligned}$$

where

$$U'_{i+1/2} = U'\left(\frac{u_{i+1} - u_i}{\Delta_{i+1/2}}\right),$$

and

$$p_{j+1/2}(u_h|_{I_m}) \quad = \quad \overline{u}_m - \frac{\delta}{\Delta_{j+1}}\, f^+(u^-_{m+1/2}) + \frac{\delta}{\Delta_j}\, f^-(u^+_{m-1/2}).$$

From the above result, we see that the total variation of the means of the Euler forward step is nonincreasing if the following three conditions are satisfied:

$$sgn(\,\overline{u}_{j+1} - \overline{u}_j\,) \quad = \quad sgn(\,p_{j+1/2}(u_h|_{I_{j+1}}) - p_{j+1/2}(u_h|_{I_j})\,), \qquad (2.3.16)$$

$$sgn(\,\overline{u}_j - \overline{u}_{j-1}\,) \quad = \quad sgn(\,u^{n,-}_{j+1/2} - u^{n,-}_{j-1/2}\,), \qquad (2.3.17)$$

$$sgn(\,\overline{u}_{j+1} - \overline{u}_j\,) \quad = \quad sgn(\,u^{n,+}_{j+1/2} - u^{n,+}_{j-1/2}\,). \qquad (2.3.18)$$

Note that if the properties (2.3.16) and (2.3.17) are satisfied, then the property (2.3.18) can always be satisfied for a small enough values of $|\,\delta\,|$.

Of course, the numerical method under consideration does not provide an approximate solution automatically satisfying the above conditions. It is thus necessary to *enforce* them by means of a suitably defined generalized slope limiter,' $\Lambda\Pi_h$.

2.4. The generalized slope limiter

2.4.1. High-order accuracy versus the TVDM property: Heuristics. The ideal generalized slope limiter $\Lambda\Pi_h$

- Maintains the conservation of *mass* element by element,
- Satifies the properties (2.3.16), (2.3.17), and (2.3.18),
- Does not degrade the accuracy of the method.

The first requirement simply states that the slope limiting must not change the total mass contained in each interval, that is, if $u_h = \Lambda\Pi_h(v_h)$,

$$\overline{u}_j = \overline{v}_j, \qquad j = 1, \ldots, N.$$

This is, of course a very sensible requirement because after all we are dealing with consevation laws. It is also a requirement very easy to satisfy.

The second requirement, states that if $u_h = \Lambda\Pi_h(v_h)$ and $w_h = u_h + \delta\, L_h(u_h)$ then

$$|\,\overline{w}_h\,|_{TV(0,1)} \leq |\,\overline{u}_h\,|_{TV(0,1)},$$

for small enough values of $|\,\delta\,|$.

The third requirement deserves a more delicate discussion. Note that if u_h is a very good approximation of a smooth solution u in a neigborhood of the point x_0, it behaves (asymptotically as Δx goes to zero) as a straight line if $u_x(x_0) \neq 0$. If x_0 is an isolated extrema of u, then it behaves like a parabola provided $u_{xx}(x_0) \neq 0$. Now, if u_h is a straightline, it trivially satisfies conditions (2.3.16) and (2.3.17). However, if u_h is a parabola, conditions (2.3.16) and (2.3.17) are not always satisfied. This shows that it is impossible to construct the above ideal generalized 'solpe limiter,' or, in other words, that in order to enforce the TVDM property, we must loose high-order accuracy at the local extrema. This is a very well-known phenomenon for TVD finite difference schemes!

Fortunatelly, it is still possible to construct generalized slope limiters that do preserve high-order accuracy even at local extrema. The resulting scheme will then not be TVDM but total variation bounded in the means (TVBM) as we will show.

In what follows we first consider generalized slope limiters that render the RKDG schemes TVDM. Then we suitably modify them in order to obtain TVBM schemes.

2.4.2. Constructing TVDM generalized slope limiters. Next, we look for simple, sufficient conditions on the function u_h that imply the conditions (2.3.16), (2.3.17), and (2.3.18). These conditions will be stated in terms of the *minmod* function m defined as follows:

$$m\,(a_1, \ldots, a_\nu) \;=\; \begin{cases} s\,\min_{1\leq n\leq\nu}\,|\,a_n\,|, & \text{if } s = sign(a_1) = \cdots = sign(a_\nu), \\ 0, & \text{otherwise.} \end{cases}$$

THEOREM 2.10. *Suppose the the following CFL condition is satisfied:*

$$|\,\delta\,|\,\Big(\frac{|\,f^+\,|_{Lip}}{\Delta_{j+1}} + \frac{|\,f^-\,|_{Lip}}{\Delta_j}\Big) \leq 1/2, \qquad j = 1, \ldots, N. \tag{2.4.19}$$

Then, conditions (2.3.16), (2.3.17), and (2.3.18) are satisfied if, for all $j = 1, \ldots, N$, *we have that*

$$\begin{aligned} u^-_{j+1/2} - \overline{u}_j &= m\,(u^-_{j+1/2} - \overline{u}_j, \overline{u}_j - \overline{u}_{j-1}, \overline{u}_{j+1} - \overline{u}_j) && (2.4.20)\\ \overline{u}_j - u^+_{j-1/2} &= m\,(\overline{u}_j - u^+_{j-1/2}, \overline{u}_j - \overline{u}_{j-1}, \overline{u}_{j+1} - \overline{u}_j). && (2.4.21) \end{aligned}$$

Proof. Let us start by showing that the property (2.3.17) is satisfied. We have:

$$\begin{aligned} u^-_{j+1/2} - u^-_{j-1/2} &= (u^-_{j+1/2} - \overline{u}_j) + (\overline{u}_j - \overline{u}_{j-1}) + (\overline{u}_{j-1} - u^-_{j-1/2})\\ &= \Theta\,(\overline{u}_j - \overline{u}_{j-1}), \end{aligned}$$

where

$$\Theta = 1 + \frac{u^-_{j+1/2} - \overline{u}_j}{\overline{u}_j - \overline{u}_{j-1}} - \frac{u^-_{j-1/2} - \overline{u}_{j-1}}{\overline{u}_j - \overline{u}_{j-1}} \in [0, 2],$$

by conditions (2.4.20) and (2.4.21). This implies that the property (2.3.17) is satisfied. Properties (2.3.18) and (2.3.16) are proven in a similar way. This completes the proof. □

2.4.3. Examples of TVDM generalized slope limiters.

a. The MUSCL limiter. In the case of piecewise linear approximate solutions, that is,

$$v_h|_{I_j} = \overline{v}_j + (x - x_j)\, v_{x,j}, \qquad j = 1, \ldots, N,$$

the following generalized slope limiter does satisfy the conditions (2.4.20) and (2.4.21):

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

This is the well-known slope limiter of the MUSCL schemes of vanLeer [**62, 63**].

b. The less restrictive limiter $\Lambda\Pi^1_h$. The following less restrictive slope limiter also satisfies the conditions (2.4.20) and (2.4.21):

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

Moreover, it can be rewritten as follows:

$$\begin{aligned} u^-_{j+1/2} &= \overline{v}_j + m\,(v^-_{j+1/2} - \overline{v}_j, \overline{v}_j - \overline{v}_{j-1}, \overline{v}_{j+1} - \overline{v}_j) && (2.4.22)\\ u^+_{j-1/2} &= \overline{v}_j - m\,(\overline{v}_j - v^+_{j-1/2}, \overline{v}_j - \overline{v}_{j-1}, \overline{v}_{j+1} - \overline{v}_j). && (2.4.23) \end{aligned}$$

We denote this limiter by $\Lambda\Pi^1_h$.

Note that we have that

$$\|\, \overline{v}_h - \Lambda\Pi^1_h(v_h)\, \|_{L^1(0,1)} \leq \frac{\Delta x}{2}\, |\, \overline{v}_h\, |_{TV(0,1)}.$$

See Theorem 2.13 below.

c. The limiter $\Lambda\Pi_h^k$. In the case in which the approximate solution is piecewise a polynomial of degree k, that is, when

$$v_h(x,t) \quad = \quad \sum_{\ell=0}^{k} v_j^\ell \, \varphi_\ell(x),$$

where

$$\varphi_\ell(x) = P_\ell(2\,(x - x_j)/\Delta_j),$$

and P_ℓ are the Legendre polynomials, we can define a generalized slope limiter in a very simple way. To do that, we need the define what could be called the P^1-part of v_h:

$$v_h^1(x,t) \quad = \quad \sum_{\ell=0}^{1} v_j^\ell \, \varphi_\ell(x),$$

We define $u_h = \Lambda\Pi_h(v_h)$ as follows:

- For $j = 1, ..., N$ compute $u_h|_{I_j}$ as follows:
  1. Compute $u_{j+1/2}^-$ and $u_{j-1/2}^+$ by using (2.4.22) and (2.4.23),
  2. If $u_{j+1/2}^- = v_{j+1/2}^-$ and $u_{j-1/2}^+ = v_{j-1/2}^+$ set $u_h|_{I_j} = v_h|_{I_j}$,
  3. If not, take $u_h|_{I_j}$ equal to $\Lambda\Pi_h^1(v_h^1)$.

d. The limiter $\Lambda\Pi_{h,\alpha}^k$. When instead of (2.4.22) and (2.4.23), we use

$$u_{j+1/2}^- \quad = \quad \overline{v}_j + m\,(v_{j+1/2}^- - \overline{v}_j, \, \overline{v}_j - \overline{v}_{j-1}, \, \overline{v}_{j+1} - \overline{v}_j, C\,(\Delta x)^\alpha) \quad (2.4.24)$$

$$u_{j-1/2}^+ \quad = \quad \overline{v}_j - m\,(\overline{v}_j - v_{j-1/2}^+, \, \overline{v}_j - \overline{v}_{j-1}, \, \overline{v}_{j+1} - \overline{v}_j, C\,(\Delta x)^\alpha), \quad (2.4.25)$$

for some fixed constant C and $\alpha \in (0,1)$, we obtain a generalized slope limiter we denote by $\Lambda\Pi_{h,\alpha}^k$.

This generalized slope limiter is never used in practice, but we consider it here because it is used for theoretical purposes; see Theorem 2.13 below.

2.4.4. The complete RKDG method. Now that we have our generalized slope limiters, we can display the complete RKDG method. It is contained in the following algorith:

- Set $u_h^0 = \Lambda\Pi_h \, P_{V_h}(u_0)$;
- For $n = 0, ..., N-1$ compute u_h^{n+1} as follows:
  1. set $u_h^{(0)} = u_h^n$;
  2. for $i = 1, ..., k+1$ compute the intermediate functions:
  $$u_h^{(i)} = \Lambda\Pi_h \left\{ \sum_{l=0}^{i-1} \alpha_{il} \, u_h^{(l)} + \beta_{il} \Delta t^n L_h(u_h^{(l)}) \right\};$$
  3. set $u_h^{n+1} = u_h^{(k+1)}$.

This algorithm describes the complete RKDG method. Note how the generalized slope limiter has to be applied at each intermediate computation of the Runge-Kutta method. This way of appying the generalized slope limiter in the time-marching algorithm ensures that the scheme is TVDM, as we next show.

2.4.5. The TVDM property of the RKDG method. To do that, we start by noting that if we set

$$u_h = \Lambda\Pi_h(v_h), \qquad w_h = u_h + \delta\, L_h(u_h),$$

then we have that

$$|\,\overline{u}_h\,|_{TV(0,1)} \;\leq\; |\,\overline{v}_h\,|_{TV(0,1)}, \tag{2.4.26}$$
$$|\,\overline{w}_h\,|_{TV(0,1)} \;\leq\; |\,\overline{u}_h\,|_{TV(0,1)}, \qquad \forall\,|\,\delta\,|\,\leq \delta_0, \tag{2.4.27}$$

where

$$\delta_0^{-1} = 2\,\max_j\,(\frac{|\,f^+\,|_{Lip}}{\Delta_{j+1}} + \frac{|\,f^-\,|_{Lip}}{\Delta_j}) \qquad j = 1, \ldots, N,$$

by Theorem 2.10. By using the above two properties of the generalized slope limiter,' it is possible to show that the RKDG method is TVDM.

THEOREM 2.11. *Assume that the generalized slope limiter $\Lambda\Pi_h$ satisfies the properties (2.4.26) and (2.4.27). Assume also that all the coeficients α_{il} are non-negative and satisfy the following condition:*

$$\sum_{l=0}^{i-1} \alpha_{il} = 1, \qquad i = 1, \ldots, k+1.$$

Then

$$|\,\overline{u}_h^n\,|_{TV(0,1)} \leq |\,u_0\,|_{TV(0,1)}, \qquad \forall\, n \geq 0.$$

Proof of Theorem 2.11. The proof of this result is very similar to the proof of Theorem 2.5. Thus, we start by rewriting our time discretization as follows:

- Set $u_h^0 = u_{0h}$;
- For $n = 0, \ldots, N-1$ compute u_h^{n+1} from u_h^n as follows:
  1. set $u_h^{(0)} = u_h^n$;
  2. for $i = 1, \ldots, k+1$ compute the intermediate functions:

     $$u_h^{(i)} = \Lambda\Pi_h \left\{ \sum_{l=0}^{i-1} \alpha_{il}\, w_h^{(il)} \right\},$$

     where

     $$w_h^{(il)} = u_h^{(l)} + \frac{\beta_{il}}{\alpha_{il}}\,\Delta t^n\, L_h(u_h^{(l)});$$

  3. set $u_h^{n+1} = u_h^{(k+1)}$.

Then have,

$$
\begin{aligned}
|\,\overline{u}_h^{(i)}\,|_{TV(0,1)} &\le |\sum_{l=0}^{i-1} \alpha_{il}\,\overline{w}_h^{(il)}\,|_{TV(0,1)}, \quad \text{by (2.4.26)},\\
&\le \sum_{l=0}^{i-1} \alpha_{il}\,|\,\overline{w}_h^{(il)}\,|_{TV(0,1)}, \quad \text{since } \alpha_{il} \ge 0,\\
&\le |\sum_{l=0}^{i-1} \alpha_{il}\,\overline{u}_h^{(l)}\,|_{TV(0,1)}, \quad \text{by (2.4.27)},\\
&\le \max_{0\le l\le i-1} |\,\overline{u}_h^{(l)}\,|_{TV(0,1)}, \quad \text{since } \sum_{l=0}^{i-1} \alpha_{il} = 1.
\end{aligned}
$$

It is clear now that that the inequality

$$|\,\overline{u}_h^{n}\,|_{TV(0,1)} \le |\,\overline{u}_h^{0}\,|_{TV(0,1)}, \qquad \forall\, n \ge 0.$$

follows from the above inequality by a simple induction argument. To obtain the result of the theorem, it is enough to note that we have

$$|\,\overline{u}_h^{0}\,|_{TV(0,1)} \le |\,u_0\,|_{TV(0,1)},$$

by the definition of the initial condition u_h^0. This completes the proof. □

2.4.6. TVBM generalized slope limiters. As was pointed out before, it is possible to modify the generalized slope limiters displayed in the examples above in such a way that the degradation of the accuracy at local extrema is avoided. To achieve this, we follow Shu [**59**] and modify the definition of the generalized slope limiters by simply replacing the *minmod* function m by the TVB corrected *minmod* function $\bar{m}$ defined as follows:

$$\bar{m}\,(a_1,...,a_m) = \begin{cases} a_1, & if\;\; |a_1| \le M(\Delta x)^2, \\ m\,(a_1,...,a_m), & otherwise, \end{cases} \tag{2.4.28}$$

where M is a given constant. We call the generalized slope limiters thus constructed, TVBM slope limiters.

The constant 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

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

See [**15**] for other choices of M.

Thus, if the constant M is is taken as above, there is no degeneracy of accuracy at the extrema and the resulting RKDG scheme retains its optimal accuracy. Moreover, we have the following stability result.

THEOREM 2.12. *Assume that the generalized slope limiter $\Lambda\Pi_h$ is a TVBM slope limiter. Assume also that all the coeficients α_{il} are nonnegative and satisfy the following condition:*

$$\sum_{l=0}^{i-1} \alpha_{il} = 1, \qquad i = 1, \ldots, k+1.$$

Then

$$|\,\overline{u}_h^n\,|_{TV(0,1)} \le |\,\overline{u}_0\,|_{TV(0,1)} + C\,M, \qquad \forall n \ge 0,$$

where C depends on k only.

2.4.7. Convergence in the nonlinear case. By using the stability above stability results, we can use the Ascoli-Arzelá theorem to prove the following convergence result.

THEOREM 2.13. *Assume that the generalized slope limiter $\Lambda\Pi_h$ is a TVDM or a TVBM slope limiter. Assume also that all the coeficients α_{il} are nonnegative and satisfy the following condition:*

$$\sum_{l=0}^{i-1} \alpha_{il} = 1, \qquad i = 1, \ldots, k+1.$$

Then there is a subsequence $\{\,\overline{u}_{h'}\}_{h'>0}$ of the sequence $\{\,\overline{u}_h\}_{h>0}$ generate by the RKDG scheme that converges in $L^\infty(0,T;L^1(0,1))$ to a weak solution of the problem (2.1.1), (2.1.2).

Moreover, if the TVBM version of the slope limiter $\Lambda\Pi^k_{h,\alpha}$ is used, the weak solution is the entropy solution and the whole sequence converges.

Finally, if the generalized slope limiter $\Lambda\Pi_h$ is such that

$$\|\,\overline{v}_h - \Lambda\Pi_h(v_h)\,\|_{L^1(0,1)} \le C\,\Delta x\,|\,\overline{v}_h\,|_{TV(0,1)},$$

then the above results hold not only to the sequence of the means $\{\,\overline{u}_h\}_{h>0}$ but to the sequence of the functions $\{\,u_h\}_{h>0}$.

2.5. Computational results

In this section, we display the performance of the RKDG schemes in a simple but typical test problem. We use piecewise linear ($k=1$) and piecewise quadratic ($k=2$) elements; the $\Lambda\Pi_h^k$ generalized slope limter is used. Our purpose is to show that (i) when the constant M is properly chosen, the RKDG method using polynomials of degree k is is order $k+1$ in the uniform norm away from the discontinuities, that (ii) it is computationally more efficient to use high-degree polynomial approximations, and that (iii) shocks are captured in a few elements without production of spurious oscillations

We solve the Burger's equation with a periodic boundary condition:

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

The exact solution is smooth at $T = .05$ and has a well developed shock at $T = 0.4$. Notice that there is a sonic point. In Tables 1,2, and 3, the history of convergence of the RKDG method using piecewise linear elements is dsplayed and in Tables 4,5, and 6, the history of convergence of the RKDG method using piecewise quadratic elements. 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 $M = 20 \ge 2\,\pi^2$, the scheme is uniformly high order in regions of smoothness that include critical and sonic points.

Next, we compare the efficiency of the RKDG schemes for $k = 1$ and $k = 2$ for the case $M = 20$ and $T = 0.05$. We define the inverse of the efficiency of the method as the product of the error times the number of operations. Since the RKDG method that uses quadratic elements has $0.3/0.2$ times more time steps, $3/2$ times more inner iterations per time step, and $3/2$ time more unknowns in space, its number of operations is $27/8$ times bigger than the one of the RKDH method using linear elements. Hence, the ratio of the efficiency of the RKDG method with quadratic elements to that of the RKDG method with linear elements is

$$r = \frac{8}{27} \frac{error(RKDG(k=1)}{error(RKDG(k=2)}.$$

The results are displayed in Table 7. We can see that the efficiency of the RKDG scheme with quadratic polynomials is several times that of the RKDG scheme with linear polynomials even for very small values of Δx. We can also see that the ratio r of efficiencies is proportional to $(\Delta x)^{-1}$, which is expected for smooth solutions. This indicates that it is indeed more efficient to work with RKDG methods using polynomials of higher degree.

That this is also true when the solution displays discontinuities can be seen figures 1, and 2. In the figure 1, it can be seen that the shock is captured in essentially two elements. A zoom of these figures is shown in figure 2, where the approximation right in front of the shock is shown. It is clear that the approximation using quadratic elements is superior to the approximation using linear elements.

2.6. Concluding remarks

In this section, which is the core of these notes, we have devised the general RKDG method for nonlinear scalar conservation laws with periodic boundary conditions.

We have seen that the RKDG are constructed in three steps. First, the Discontinuous Galerkin method is used to discretize in space the conservation law. Then, an explicit TVB-Runge-Kutta time discretizationis used to discretize the resulting ODE system. Finally, a generalized slope limiter is introduced that enforces nonlinear stability without degrading the accuracy of the scheme.

We have seen that the numerical results show that the RKDG methods using polynomials of degree $k, k = 1, 2$ are uniformly $(k+1)$-th order accurate away from discontinuities and that the use of high degree polynomials render the RKDG method more efficient, even close to discontinuities.

All these results can be extended to the initial boundary value problem, see [**15**]. In what follows, we extend the RKDG methods to multidimensional systems.

Table 1
P^1, $M = 0$, CFL= 0.3, $T = 0.05$.

	$L^1(0,1) - error$		$L^\infty(0,1) - error$	
Δx	$10^5 \cdot error$	order	$10^5 \cdot error$	order
1/10	1286.23	-	3491.79	-
1/20	334.93	1.85	1129.21	1.63
1/40	85.32	1.97	449.29	1.33
1/80	21.64	1.98	137.30	1.71
1/160	5.49	1.98	45.10	1.61
1/320	1.37	2.00	14.79	1.61
1/640	0.34	2.01	4.85	1.60
1/1280	0.08	2.02	1.60	1.61

Table 2
P^1, $M = 20$, CFL= 0.3, $T = 0.05$.

	$L^1(0,1) - error$		$L^\infty(0,1) - error$	
Δx	$10^5 \cdot error$	order	$10^5 \cdot error$	order
1/10	1073.58	-	2406.38	-
1/20	277.38	1.95	628.12	1.94
1/40	71.92	1.95	161.65	1.96
1/80	18.77	1.94	42.30	1.93
1/160	4.79	1.97	10.71	1.98
1/320	1.21	1.99	2.82	1.93
1/640	0.30	2.00	0.78	1.86
1/1280	0.08	2.00	0.21	1.90

Table 3
Errors in smooth region $\Omega = \{x : |x - shock| \geq 0.1\}$.
P^1, $M = 20$, CFL$= 0.3$, $T = 0.4$.

	$L^1(\Omega) - error$		$L^\infty(\Omega) - error$	
Δx	$10^5 \cdot error$	order	$10^5 \cdot error$	order
1/10	1477.16	-	17027.32	-
1/20	155.67	3.25	1088.55	3.97
1/40	38.35	2.02	247.35	2.14
1/80	9.70	1.98	65.30	1.92
1/160	2.44	1.99	17.35	1.91
1/320	0.61	1.99	4.48	1.95
1/640	0.15	2.00	1.14	1.98
1/1280	0.04	2.00	0.29	1.99

Table 4
P^2, $M = 0$, CFL$= 0.2$, $T = 0.05$.

	$L^1(0,1) - error$		$L^\infty(0,1) - error$	
Δx	$10^5 \cdot error$	order	$10^5 \cdot error$	order
1/10	2066.13	-	16910.05	-
1/20	251.79	3.03	3014.64	2.49
1/40	42.52	2.57	1032.53	1.55
1/80	7.56	2.49	336.62	1.61

Table 5

P^2, $M = 20$, CFL$= 0.2$, $T = 0.05$.

	$L^1(0,1) - error$		$L^\infty(0,1) - error$	
Δx	$10^5 \cdot error$	order	$10^5 \cdot error$	order
1/10	37.31	-	101.44	-
1/20	4.58	3.02	13.50	2.91
1/40	0.55	3.05	1.52	3.15
1/80	0.07	3.08	0.19	3.01

Table 6

Errors in smooth region $\Omega = \{x : |x - shock| \geq 0.1\}$.

P^2, $M = 20$, CFL$= 0.2$, $T = 0.4$.

	$L^1(\Omega) - error$		$L^\infty(\Omega) - error$	
Δx	$10^5 \cdot error$	order	$10^5 \cdot error$	order
1/10	786.36	-	16413.79	-
1/20	5.52	7.16	86.01	7.58
1/40	0.36	3.94	15.49	2.47
1/80	0.06	2.48	0.54	4.84

Table 7

Comparison of the efficiencies of RKDG schemes for $k = 2$ and $k = 1$

$M = 20$, $T = 0.05$.

	L^1-norm		L^∞-norm	
Δx	$eff.ratio$	order	$eff.ratio$	order
1/10	8.52	-	7.03	-
1/20	17.94	-1.07	46.53	-2.73
1/40	38.74	-1.11	106.35	-1.19
1/80	79.45	-1.04	222.63	-1.07

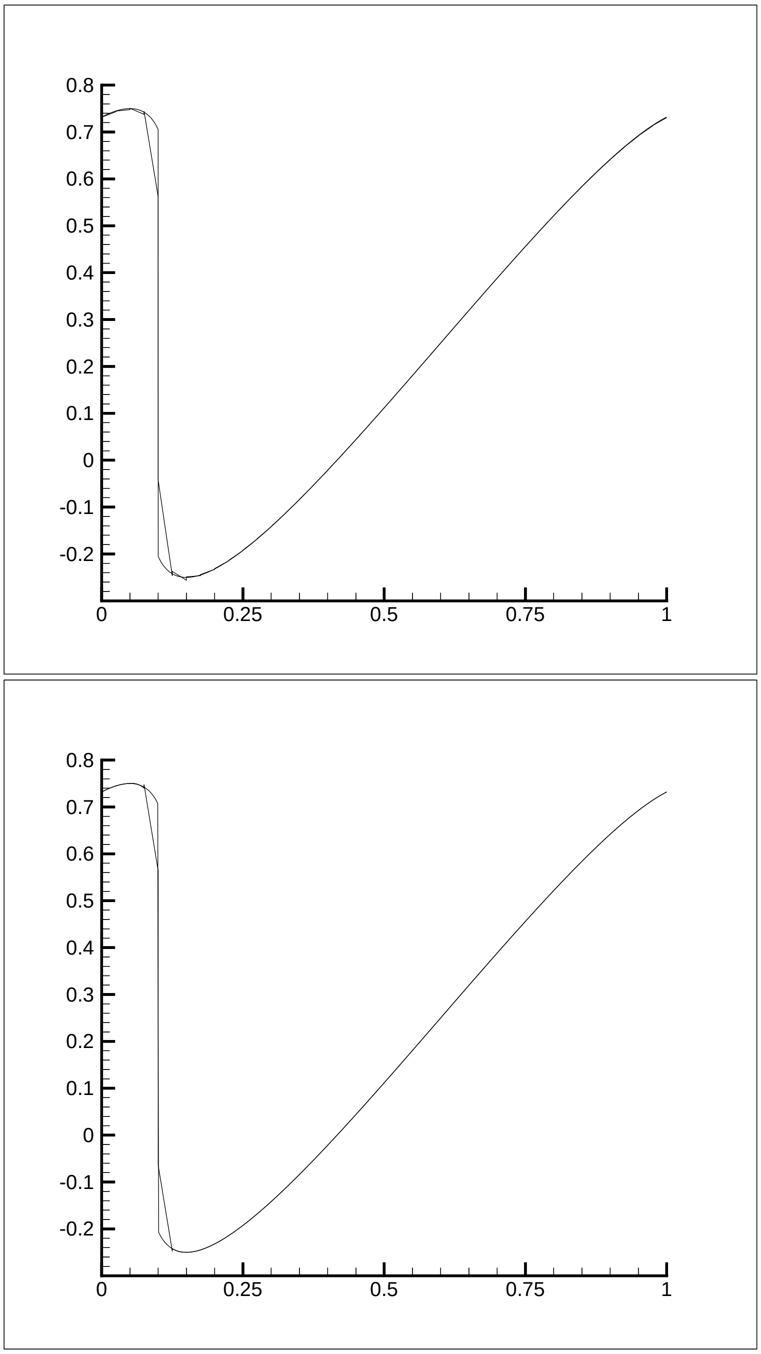


FIGURE 1. Comparison of the exact and the approximate solution obtained with $M = 20$, $\Delta x = 1/40$ at $T = .4$: Piecewise linear elements (top) and piecewise quadratic elements (bottom)

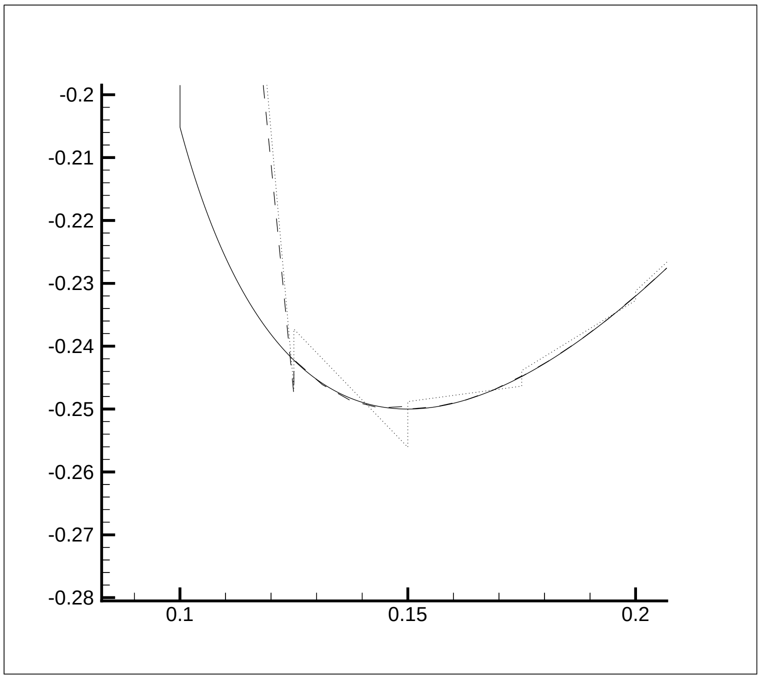


FIGURE 2. Detail of previous figure. Behavior of the approximate solutions four elements in front of the shock: Exact solution (solid line), piecewise linear solution (dotted line), and piecewise quadratic solution (dashed line).

2.7. Appendix: Proof of the $\mathbf{L}^2$-error estimates in the linear case

2.7.1. Proof of the $\mathbf{L}^2$-stability. In this section, we prove the the stability result of Proposition 2.1. To do that, we first show how to obtain the corresponding stability result for the exact solution and then mimic the argument to obtain Proposition 2.1.

The continuous case as a model. We start by rewriting the equations (2.2.4) in *compact form.* If in the equations (2.2.4) we replace $v(x)$ by $v(x,t)$, sum on j from 1 to N, and integrate in time from 0 to T, we obtain

$$\mathbb{B}(u,v) \quad = \quad 0, \qquad \forall\ v : v(t) \text{ is smooth} \quad \forall\ t \in (0,T), \tag{2.7.29}$$

where

$$\mathbb{B}(u,v) \quad = \quad \int_0^T \int_0^1 \left\{\, \partial_t u(x,t)\, v(x,t) - c\, u(x,t)\, \partial_x\, v(x,t) \,\right\} dx\, dt. \tag{2.7.30}$$

Taking $v = u$, we easily see that we see that

$$\mathbb{B}(u,u) = \frac{1}{2} \|\, u(T)\, \|^2_{L^2(0,1)} - \frac{1}{2} \|\, u_0\, \|^2_{L^2(0,1)},$$

and since

$$\mathbb{B}(u,u) = 0,$$

by (2.7.29), we immediately obtain the following L^2-stability result:

$$\frac{1}{2}\| u(T) \|^2_{L^2(0,1)} = \frac{1}{2}\| u_0 \|^2_{L^2(0,1)}.$$

This is the argument we have to mimic in order to prove Proposition 2.1.

The discrete case. Thus, we start by finding the discrete version of the form $\mathbb{B}(\cdot,\cdot)$. If we replace $v(x)$ by $v_h(x,t)$ in the equation (2.2.7), sum on j from 1 to N, and integrate in time from 0 to T, we obtain

$$\mathbb{B}_h(u_h, v_h) = 0, \qquad \forall\, v_h : \, v_h(t) \in V_h^k \quad \forall\, t \in (0,T). \tag{2.7.31}$$

where

$$\begin{aligned} \mathbb{B}_h(u_h, v_h) \quad &= \int_0^T \int_0^1 \partial_t u_h(x,t)\, v_h(x,t)\, dx\, dt \\ &- \int_0^T \sum_{1\le j\le N} \int_{I_j} c\, u_h(x,t)\, \partial_x\, v_h(x,t)\, dx\, dt \\ &- \int_0^T \sum_{1\le j\le N} h(u_h)_{j+1/2}(t)\, [\, v_h(t)\,]_{j+1/2}\, dt. \end{aligned} \tag{2.7.32}$$

Following the model provided by the continuous case, we next obtain an expression for $\mathbb{B}_h(w_h, w_h)$. It is contained in the following result which will proved later.

Lemma 2.14. *We have*

$$\mathbb{B}_h(w_h, w_h) = \frac{1}{2}\| w_h(T) \|^2_{L^2(0,1)} + \Theta_T(w_h) - \frac{1}{2}\| w_h(0) \|^2_{L^2(0,1)},$$

where

$$\Theta_T(\, w_h\,) = \tfrac{|\,c\,|}{2} \int_0^T \textstyle\sum_{1\le j\le N} [\, w_h(t)\,]^2_{j+1/2}\, dt.$$

Taking $w_h = u_h$ in the above result and noting that by (2.7.31),

$$\mathbb{B}_h(u_h, u_h) = 0,$$

we get the equality

$$\tfrac{1}{2}\| u_h(T) \|^2_{L^2(0,1)} + \Theta_T(u_h) = \tfrac{1}{2}\| u_h(0) \|^2_{L^2(0,1)},$$

from which Proposition 2.1 easily follows, since

$$\frac{1}{2}\| u_h(T) \|^2_{L^2(0,1)} \quad \le \quad \frac{1}{2}\| u_0 \|^2_{L^2(0,1)},$$

by (2.2.8). It only remains to prove Lemma 2.14.

Proof of Lemma 2.14. After setting $u_h = v_h = w_h$ in the definition of $\mathbb{B}_h$, (2.7.32), we get

$$\mathbb{B}_h(w_h, w_h) \quad = \quad \frac{1}{2}\| w_h(T) \|^2_{L^2(0,1)} + \int_0^T \Theta_{diss}(t)\, dt - \frac{1}{2}\| w_h(0) \|^2_{L^2(0,1)},$$

where

$$\Theta_{diss}(t) \quad = \quad - \sum_{1\le j\le N} \left\{ h(w_h)_{j+1/2}(t)\,[\,w_h(t)\,]_{j+1/2} + \int_{I_j} c\,w_h(x,t)\,\partial_x\, w_h(x,t)\,dx \right\}.$$

We only have to show that $\int_0^T \Theta_{diss}(t)\,dt = \Theta_T(w_h)$. To do that, we proceed as follows. Dropping the dependence on the variable t and setting

$$\overline{w}_h(x_{j+1/2}) = \frac{1}{2}(\,w_h(x^-_{j+1/2}) + w_h(x^+_{j+1/2})\,),$$

we have, by the definition of the flux h, (2.2.11),

$$- \sum_{1\le j\le N} \int_{I_j} h(w_h)_{j+1/2}\,[\,w_h\,]_{j+1/2} \quad = \quad - \sum_{1\le j\le N} \{\,c\,\overline{w}_h\,[\,w_h\,] - \frac{|\,c\,|}{2}\,[\,w_h\,]^2\,\}_{j+1/2},$$

and

$$\begin{aligned} - \sum_{1\le j\le N} \int_{I_j} c\,w_h(x)\,\partial_x\, w_h(x)\,dx \quad &= \quad \frac{c}{2} \sum_{1\le j\le N} [\,w_h^2\,]_{j+1/2} \\ &= \quad c \sum_{1\le j\le N} \{\overline{w}_h\,[\,w_h\,]\}_{j+1/2} \end{aligned}$$

Hence

$$\Theta_{diss}(t) \quad = \quad \frac{|\,c\,|}{2} \sum_{1\le j\le N} [u_h(t)]^2_{j+1/2},$$

and the result follows. This completes the proof of Lemma 2.14. □

This completes the proof of Proposition 2.1.

2.7.2. Proof of the Theorem 2.1. In this section, we prove the error estimate of Theorem 2.1 which holds for the linear case $f(u) = c\,u$. To do that, we first show how to estimate the error between the solutions $w_\nu = (u_\nu, q_\nu)^t$, $\nu = 1, 2$, of

$$\begin{aligned} &\partial_t\, u_\nu + \partial_x\, f(u_\nu) = 0 \quad \text{in } (0,T)\times(0,1), \\ &u_\nu(t=0) = u_{0,\nu}, \quad \text{on } (0,1). \end{aligned}$$

Then, we mimic the argument in order to prove Theorem 2.1.

The continuous case as a model. By the definition of the form $\mathbb{B}(\cdot,\cdot)$, (2.7.30), we have, for $\nu = 1, 2$,

$$\mathbb{B}(w_\nu, v) \quad = \quad 0, \qquad \forall\; v : v(t) \text{ is smooth} \quad \forall\; t \in (0,T).$$

Since the form $\mathbb{B}(\cdot,\cdot)$ is bilinear, from the above equation we obtain the so-called *error equation*:

$$\mathbb{B}(e, v) \quad = \quad 0, \qquad \forall\; v : v(t) \text{ is smooth} \quad \forall\; t \in (0,T). \tag{2.7.33}$$

where $e = w_1 - w_2$. Now, since

$$\mathbb{B}(e, e) \quad = \quad \frac{1}{2}\|\,e(T)\,\|^2_{L^2(0,1)} - \frac{1}{2}\|\,e(0)\,\|^2_{L^2(0,1)},$$

and

$$\mathbb{B}(e, e) = 0,$$

by the error equation (2.7.33), we immediately obtain the error estimate we sought:

$$\frac{1}{2}\| e(T) \|^2_{L^2(0,1)} = \frac{1}{2}\| u_{0,1} - u_{0,2} \|^2_{L^2(0,1)}.$$

To prove Theorem 2.1, we only need to obtain a discrete version of this argument.

The discrete case. Since,

$$\begin{aligned} \mathbb{B}_h(u_h, v_h) = 0, \qquad &\forall\, v_h :\ v(t) \in V_h \quad \forall\, t \in (0,T), \\ \mathbb{B}_h(u, v_h) = 0, \qquad &\forall\, v_h :\ v_h(t) \in V_h \quad \forall\, t \in (0,T), \end{aligned}$$

by (2.2.7) and by equations (2.2.4), respectively, we easily obtain our *error equation*:

$$\mathbb{B}_h(e, v_h) = 0, \qquad \forall\, v_h :\ v_h(t) \in V_h \quad \forall\, t \in (0,T), \tag{2.7.34}$$

where $e = w - w_h$.

Now, according to the continuous case argument, we should consider next the quantity $\mathbb{B}_h(e,e)$; however, since $e(t)$ is not in the finite element space V_h, it is more convenient to consider $\mathbb{B}_h(\mathbb{P}_h(e), \mathbb{P}_h(e))$, where $\mathbb{P}_h(e(t))$ is the L^2-projection of the error $e(t)$ into the finite element space V_h^k.

The L^2-projection of the function $p \in \mathrm{L}^2(0,1)$ into V_h, $\mathbb{P}_h(p)$, is defined as the only element of the finite element space V_h such that

$$\textstyle\int_0^1 \big(\, \mathbb{P}_h(p)(x) - p(x) \,\big)\, v_h(x)\, dx = 0, \qquad \forall\ v_h \in V_h. \tag{2.7.35}$$

Note that in fact $u_h(t=0) = \mathbb{P}_h(u_0)$, by (2.2.8).

Thus, by Lemma 2.14, we have

$$\mathbb{B}_h(\mathbb{P}_h(e), \mathbb{P}_h(e)) \;=\; \frac{1}{2}\| \mathbb{P}_h(e(T)) \|^2_{L^2(0,1)} + \Theta_T(\mathbb{P}_h(e)) - \frac{1}{2}\| \mathbb{P}_h(e(0)) \|^2_{L^2(0,1)},$$

and since

$$\mathbb{P}_h(e(0)) = \mathbb{P}_h(u_0 - u_h(0)) = \mathbb{P}_h(u_0) - u_h(0) = 0,$$

and

$$\mathbb{B}_h(\mathbb{P}_h(e), \mathbb{P}_h(e)) = \mathbb{B}_h(\mathbb{P}_h(e) - e, \mathbb{P}_h(e)) = \mathbb{B}_h(\mathbb{P}_h(u) - u, \mathbb{P}_h(e)),$$

by the *error equation* (2.7.34), we get

$$\frac{1}{2}\| \mathbb{P}_h(e(T)) \|^2_{L^2(0,1)} + \Theta_T(\mathbb{P}_h(e)) \;=\; \mathbb{B}_h(\mathbb{P}_h(u) - u, \mathbb{P}_h(e)). \tag{2.7.36}$$

It only remains to estimate the right-hand side

$$\mathbb{B}(\mathbb{P}_h(u) - u, \mathbb{P}_h(e)),$$

which, according to our continuous model, should be small.

Estimating the right-hand side. To show that this is so, we must suitably treat the term $\mathbb{B}(\mathbb{P}_h(w) - w, \mathbb{P}_h(e))$. We start with the following remarkable result.

LEMMA 2.15. *We have*

$$\mathbb{B}_h(\mathbb{P}_h(u) - u, \mathbb{P}_h(e)) \;=\; -\int_0^T \sum_{1 \le j \le N} h(\mathbb{P}_h(u) - u)_{j+1/2}(t)\, [\, \mathbb{P}_h(e)(t)\,]_{j+1/2}\, dt.$$

Proof Setting $p = \mathbb{P}_h(u) - u$ and $v_h = \mathbb{P}_h(e)$ and recalling the definition of $\mathbb{B}_h(\cdot,\cdot)$, (2.7.32), we have

$$\begin{aligned}
\mathbb{B}_h(p, v_h) &= \int_0^T \int_0^1 \partial_t p(x,t)\, v_h(x,t)\, dx\, dt \\
&\quad - \int_0^T \sum_{1\le j\le N} \int_{I_j} c\, p(x,t)\, \partial_x\, v_h(x,t)\, dx\, dt \\
&\quad - \int_0^T \sum_{1\le j\le N} h(p)_{j+1/2}(t)\, [\, v_h(t)\,]_{j+1/2}\, dt \\
&= - \int_0^T \sum_{1\le j\le N} h(p)_{j+1/2}(t)\, [\, v_h(t)\,]_{j+1/2}\, dt,
\end{aligned}$$

by the definition of the L^2-projection (2.7.35). This completes the proof. □

Now, we can see that a simple application of Young's inequality and a standard approximation result should give us the estimate we were looking for. The approximation result we need is the following.

LEMMA 2.16. *If $w \in H^{k+1}(I_j \cup I_{j+1})$, then*

$$|\, h(\mathbb{P}_h(w) - w)(x_{j+1/2})\,| \le c_k\, (\Delta x)^{k+1/2}\, \frac{|\, c\,|}{2}\, |\, w\,|_{H^{k+1}(I_j\cup I_{j+1})},$$

where the constant c_k depends solely on k.

Proof. Dropping the argument $x_{j+1/2}$ we have, by the definition (2.2.11) of the flux h,

$$\begin{aligned}
|\, h(\mathbb{P}(w) - w)\,| &= \frac{c}{2}(\mathbb{P}_h(w)^+ + \mathbb{P}_h(w)^-) - \frac{|\, c\,|}{2}(\mathbb{P}_h(w)^+ - \mathbb{P}_h(w)^-) - c\, w \\
&= \frac{c - |\, c\,|}{2}(\mathbb{P}_h(w)^+ - w) + \frac{c + |\, c\,|}{2}(\mathbb{P}_h(w)^- - w) \\
&\le |\, c\,| \max\{\, |\, \mathbb{P}_h(w)^+ - w\,|, |\, \mathbb{P}_h(w)^- - w\,|\,\}
\end{aligned}$$

and the result follows from the properties of $\mathbb{P}_h$ after a simple application of the Bramble-Hilbert lemma; see [**11**]. This completes the proof. □

An immediate consequence of this result is the estimate we wanted.

LEMMA 2.17. *We have*

$$\mathbb{B}_h(\mathbb{P}_h(u) - u, \mathbb{P}_h(e)) \le c_k^2\, (\Delta x)^{2k+1}\, \frac{|\, c\,|}{2}\, T\, |\, u_0\,|^2_{H^{k+1}(0,1)} + \frac{1}{2}\, \Theta_T(\mathbb{P}_h(e)),$$

where the constant c_k depends solely on k.

Proof. After using Young's inequality in the right-hand side of Lemma 2.15, we get

$$\begin{aligned}
\mathbb{B}_h(\mathbb{P}_h(u) - u, \mathbb{P}_h(e)) &\le \int_0^T \sum_{1\le j\le N} \frac{1}{|\, c\,|}\, |\, h(\mathbb{P}_h(u) - u)_{j+1/2}(t)\,|^2 \\
&\quad + \int_0^T \sum_{1\le j\le N} \frac{|\, c\,|}{4} [\, \mathbb{P}_h(e)(t)\,]^2_{j+1/2}\, dt.
\end{aligned}$$

By Lemma 2.16 and the definition of the form Θ_T, we get

$$\begin{aligned}\mathbb{B}_h(\mathbb{P}_h(u)-u,\mathbb{P}_h(e)) &\le c_k^2\,(\Delta x)^{2k+1}\,\frac{|\,c\,|}{4}\int_0^T \sum_{1\le j\le N} |\,u\,|^2_{H^{k+1}(I_j\cup I_{j+1})} + \frac{1}{2}\,\Theta_T(\mathbb{P}_h(e))\\ &\le c_k^2\,(\Delta x)^{2k+1}\,\frac{|\,c\,|}{2}\,T\,|\,u_0\,|^2_{H^{k+1}(0,1)} + \frac{1}{2}\,\Theta_T(\mathbb{P}_h(e)).\end{aligned}$$

This completes the proof. □

Conclusion. Finally, inserting in the equation (2.7.36) the estimate of its right hand side obtained in Lemma 2.17, we get

$$\|\,\mathbb{P}_h(e(T))\,\|^2_{L^2(0,1)} + \Theta_T(\mathbb{P}_h(e)) \le c_k\,(\Delta x)^{2k+1}\,|\,c\,|\,T\,|\,u_0\,|^2_{H^{k+1}(0,1)},$$

Theorem 2.1 now follows from the above estimate and from the following inequality:

$$\begin{aligned}\|\,e(T)\,\|_{L^2(0,1)} &\le \|\,u(T)-\mathbb{P}_h(u(T))\,\|_{L^2(0,1)} + \|\,\mathbb{P}_h(e(T))\,\|_{L^2(0,1)}\\ &\le c'_k\,(\Delta x)^{k+1}\,|\,u_0\,|_{H^{k+1}(0,1)} + \|\,\mathbb{P}_h(e(T))\,\|_{L^2(0,1)}.\end{aligned}$$

2.7.3. Proof of the Theorem 2.2. To prove Theorem 2.2, we only have to suitably modify the proof of Theorem 2.1. The modification consists in *replacing* the L^2-projection of the error, $\mathbb{P}_h(e)$, by another projection that we denote by $\mathbb{R}_h(e)$.

Given a function $p\in L^\infty(0,1)$ that is continuous on each element I_j, we define $\mathbb{R}_h(p)$ as the only element of the finite element space V_h such that

$$\forall\, j=1,\dots,N: \qquad \mathbb{R}_h(p)(x_{j,\ell}) - p(x_{j,\ell}) = 0, \qquad \ell=0,\dots,k, \tag{2.7.37}$$

where the points $x_{j,\ell}$ are the Gauss-Radau quadrature points of the interval I_j. We take

$$x_{j,k} = x_{j+1/2}, \quad \text{if } c>0, \quad \text{and} \quad x_{j,0}=x_{j-1/2}, \quad \text{if } c<0. \tag{2.7.38}$$

The special nature of the Gauss-Radau quadrature points is captured in the following property:

$$\begin{aligned}&\forall\,\varphi\in P^\ell(I_j), \quad \ell\le k, \quad \forall\, p\in P^{2k-\ell}(I_j):\\ &\int_{I_j} (\mathbb{R}_h(p)(x)-p(x))\,\varphi(x)\,dx = 0.\end{aligned} \tag{2.7.39}$$

Compare this equality with (2.7.35).

The quantity $\mathbb{B}_h(\mathbb{R}_h(e),\mathbb{R}_h(e))$. To prove our error estimate, we start by considering the quantity $\mathbb{B}_h(\mathbb{R}_h(e),\mathbb{R}_h(e))$. By Lemma 2.14, we have

$$\mathbb{B}_h(\mathbb{R}_h(e),\mathbb{R}_h(e)) = \frac{1}{2}\|\,\mathbb{R}_h(e(T))\,\|^2_{L^2(0,1)} + \Theta_T(\mathbb{R}_h(e)) - \frac{1}{2}\|\,\mathbb{R}_h(e(0))\,\|^2_{L^2(0,1)},$$

and since

$$\mathbb{B}_h(\mathbb{R}_h(e),\mathbb{R}_h(e)) = \mathbb{B}_h(\mathbb{R}_h(e)-e,\mathbb{R}_h(e)) = \mathbb{B}_h(\mathbb{R}_h(u)-u,\mathbb{R}_h(e)),$$

by the *error equation* (2.7.34), we get

$$\frac{1}{2}\| \mathbb{R}_h(e(T)) \|^2_{L^2(0,1)} + \Theta_T(\mathbb{R}_h(e)) = \frac{1}{2}\| \mathbb{R}_h(e(0)) \|^2_{L^2(0,1)} + \mathbb{B}_h(\mathbb{R}_h(u) - u, \mathbb{R}_h(e)).$$

Next, we estimate the term $\mathbb{B}(\mathbb{R}_h(u) - u, \mathbb{R}_h(e))$.

Estimating $\mathbb{B}(\mathbb{R}_h(u) - u, \mathbb{R}_h(e))$. The following result corresponds to Lemma 2.15.

LEMMA 2.18. *We have*

$$\begin{aligned} \mathbb{B}_h(\mathbb{R}_h(u) - u, v_h) &= \int_0^T \int_0^1 (\mathbb{R}_h(\partial_t u)(x,t) - \partial_t u(x,t))\, v_h(x,t)\, dx\, dt \\ &\quad - \int_0^T \sum_{1\le j\le N} \int_{I_j} c\,(\mathbb{R}_h(u)(x,t) - u(x,t))\, \partial_x\, v_h(x,t)\, dx\, dt. \end{aligned}$$

Proof Setting $p = \mathbb{R}_h(u) - u$ and $v_h = \mathbb{R}_h(e)$ and recalling the definition of $\mathbb{B}_h(\cdot,\cdot)$, (2.7.32), we have

$$\begin{aligned} \mathbb{B}_h(p, v_h) &= \int_0^T \int_0^1 \partial_t p(x,t)\, v_h(x,t)\, dx\, dt \\ &\quad - \int_0^T \sum_{1\le j\le N} \int_{I_j} c\, p(x,t)\, \partial_x\, v_h(x,t)\, dx\, dt \\ &\quad - \int_0^T \sum_{1\le j\le N} h(p)_{j+1/2}(t)\, [\, v_h(t)\,]_{j+1/2}\, dt. \end{aligned}$$

But, from the definition (2.2.11) of the flux h, we have

$$\begin{aligned} h(\mathbb{R}(u) - u) &= \frac{c}{2}(\mathbb{R}_h(u)^+ + \mathbb{R}_h(u)^-) - \frac{|\,c\,|}{2}(\mathbb{R}_h(u)^+ - \mathbb{R}_h(u)^-) - c\,u \\ &= \frac{c - |\,c\,|}{2}(\mathbb{R}_h(u)^+ - u) + \frac{c + |\,c\,|}{2}(\mathbb{R}_h(u)^- - u) \\ &= 0, \end{aligned}$$

by (2.7.38) and the result follows. ☐

Next, we need some approximation results.

LEMMA 2.19. *If* $w \in H^{k+2}(I_j)$, *and* $v_h \in P^k(I_j)$, *then*

$$\left| \int_{I_j} (\mathbb{R}_h(w) - w)(x)\, v_h(x)\, dx \right| \le c_k\, (\Delta x)^{k+1}\, |\, w\, |_{H^{k+1}(I_j)}\, \|\, v_h\, \|_{L^2(I_j)},$$

$$\left| \int_{I_j} (\mathbb{R}_h(w) - w)(x)\, \partial_x\, v_h(x)\, dx \right| \le c_k\, (\Delta x)^{k+1}\, |\, w\, |_{H^{k+2}(I_j)}\, \|\, v_h\, \|_{L^2(I_j)},$$

where the constant c_k *depends solely on* k.

Proof. The first inequality follows from the property (2.7.39) with $\ell = k$ and from standard approximation results. The second follows in a similar way from the property 2.7.39 with $\ell = k - 1$ and a standard scaling argument. This completes the proof. ☐

An immediate consequence of this result is the estimate we wanted.

LEMMA 2.20. *We have*

$$\mathbb{B}_h(\mathbb{R}_h(u)-u,\mathbb{R}_h(e)) \leq c_k\,(\Delta x)^{k+1}\,|\,u_0\,|_{H^{k+2}(0,1)} \int_0^T \|\,\mathbb{R}_h(e(t))\,\|_{L^2(0,1)}\,dt,$$

where the constant c_k depends solely on k and $|\,c\,|$.

Conclusion. Finally, inserting in the equation (2.7.36) the estimate of its right hand side obtained in Lemma 2.20, we get

$$\begin{aligned}\|\,\mathbb{R}_h(e(T))\,\|^2_{L^2(0,1)} \quad + \quad & \Theta_T(\mathbb{R}_h(e)) \leq \|\,\mathbb{R}_h(e(0))\,\|^2_{L^2(0,1)} \\ & + c_k\,(\Delta x)^{k+1}\,|\,u_0\,|_{H^{k+2}(0,1)} \int_0^T \|\,\mathbb{R}_h(e(t))\,\|_{L^2(0,1)}\,dt.\end{aligned}$$

After applying a simple variation of the Gronwall lemma, we obtain

$$\begin{aligned}\|\,\mathbb{R}_h(e(T))\,\|_{L^2(0,1)} &\leq \|\,\mathbb{R}_h(e(0))(x)\,\|_{L^2(0,1)} + c_k\,(\Delta x)^{k+1}\,T\,|\,u_0\,|_{H^{k+2}(0,1)} \\ &\leq c'_k(\Delta x)^{k+1}\,|\,u_0\,|_{H^{k+2}(0,1)}.\end{aligned}$$

Theorem 2.2 now follows from the above estimate and from the following inequality:

$$\begin{aligned}\|\,e(T)\,\|_{L^2(0,1)} &\leq \|\,u(T)-\mathbb{R}_h(u(T))\,\|_{L^2(0,1)} + \|\,\mathbb{R}_h(e(T))\,\|_{L^2(0,1)} \\ &\leq c'_k\,(\Delta x)^{k+1}\,|\,u_0\,|_{H^{k+1}(0,1)} + \|\,\mathbb{R}_h(e(T))\,\|_{L^2(0,1)}.\end{aligned}$$

CHAPTER 3

The RKDG method for multidimensional systems

3.1. Introduction

In this section, we extend the RKDG methods to multidimensional systems:

$$u_t + \nabla f(u) = 0, \qquad \text{in } \Omega \times (0,T), \tag{3.1.1}$$
$$u(x,0) = u_0(x), \qquad \forall\, x \in \Omega, \tag{3.1.2}$$

and periodic boundary conditions. For simplicity, we assume that Ω is the unit cube.

This section is essentially devoted to the description of the algorithms and their implementation details. The practitioner should be able to find here all the necessary information to completely code the RKDG methods.

This section also contains two sets of numerical results for the Euler equations of gas dynamics in two space dimensions. The first set is devoted to transient computations and domains that have corners; the effect of using triangles or rectangles and the effect of using polynomials of degree one or two are explored. The main conclusions from these computations are that (i) the RKDG method works as well with triangles as it does with rectangles and that (ii) the use of high-order polynomials does not deteriorate the approximation of strong shocks and is advantageous in the approximation of contact discontinuities.

The second set concerns steady state computations with smooth solutions. For these computations, no generalized slope limiter is needed. The effect of (i) the quality of the approximation of curved boundaries and of (ii) the degree of the polynomials on the quality of the approximate solution is explored. The main conclusions from these computations are that (i) a high-order approximation of the curve boundaries introduces a dramatic improvement on the quality of the solution and that (ii) the use of high-degree polynomials is advantageous when smooth solutions are shought.

This section contains material from the papers [**14**], [**13**], and [**19**]. It also contains numerical results from the paper by Bassi and Rebay [**2**] in two dimensions and from the paper by Warburton, Lomtev, Kirby and Karniadakis [**65**] in three dimensions.

3.2. The general RKDG method

The RKDG method for multidimensional systems has the same structure it has for one-dimensional scalar conservation laws, that is,

- Set $u_h^0 = \Lambda\Pi_h\, P_{V_h}(u_0)$;
- For $n = 0, ..., N-1$ compute u_h^{n+1} as follows:

1. set $u_h^{(0)} = u_h^n$;
2. for $i = 1, ..., k+1$ compute the intermediate functions:
$$u_h^{(i)} = \Lambda\Pi_h \left\{ \sum_{l=0}^{i-1} \alpha_{il} u_h^{(l)} + \beta_{il} \Delta t^n L_h(u_h^{(l)}) \right\};$$
3. set $u_h^{n+1} = u_h^{(k+1)}$.

In what follows, we describe the operator L_h that results form the DG-space discretization, and the generalized slope limiter $\Lambda\Pi_h$.

3.2.1. The Discontinuous Galerkin space discretization. To show how to discretize in space by the DG method, it is enough to consider the case in which u is a scalar quantity since to deal with the general case in which u, we apply the same procedure component by component.

Once a triangulation $\mathbb{T}_h$ of Ω has been obtained, we determine $L_h(\cdot)$ as follows. First, we multiply (3.1.1) by v_h in the finite elemen space V_h, integrate over the element K of the triangulation $\mathbb{T}_h$ and replace the exact solution u by its approximation $u_h \in V_h$:
$$\frac{d}{dt} \int_K u_h(t,x)\, v_h(x)\, dx + \int_K div\, f(u_h(t,x))\, v_h(x)\, dx = 0, \ \forall v_h \in V_h.$$
Integrating by parts formally we obtain
$$\begin{aligned} &\tfrac{d}{dt} \textstyle\int_K u_h(t,x)\, v_h(x)\, dx + \sum_{e\in\partial K} \int_e f(u_h(t,x)) \cdot n_{e,K}\ v_h(x)\, d\Gamma \\ &\quad - \textstyle\int_K f(u_h(t,x)) \cdot \text{ grad } v_h(x)\, dx = 0, \quad \forall v_h \in V_h, \end{aligned}$$
where $n_{e,K}$ is the outward unit normal to the edge e. Notice that $f(u_h(t,x)) \cdot n_{e,K}$ does not have a precise meaning, for u_h is discontinuous at $x \in e \in \partial K$. Thus, as in the one dimensional case, we replace $f(u_h(t,x)) \cdot n_{e,K}$ by the function $h_{e,K}(u_h(t,x^{int(K)}), u_h(t,x^{ext(K)}))$. The function $h_{e,K}(\cdot,\cdot)$ is any consistent two–point monotone Lipschitz flux, consistent with $f(u) \cdot n_{e,K}$.

In this way we obtain
$$\begin{aligned} &\tfrac{d}{dt} \textstyle\int_K u_h(t,x) v_h(x)\, dx + \sum_{e\in\partial K} \int_e h_{e,K}(t,x)\ v_h(x)\, d\,\Gamma \\ &\quad - \textstyle\int_K f(u_h(t,x)) \cdot \text{ grad } v_h(x)\, dx = 0, \qquad \forall\ v_h \in V_h. \end{aligned}$$
Finally, we replace the integrals by quadrature rules that we shall choose as follows:
$$\textstyle\int_e h_{e,K}(t,x)\, v_h(x)\, d\,\Gamma \approx \sum_{l=1}^{L} \omega_l\, h_{e,K}(t,x_{el})\, v(x_{el})|e|, \tag{3.2.3}$$
$$\begin{aligned} &\textstyle\int_K f(u_h(t,x)) \cdot \text{ grad } v_h(x)\, dx \approx \\ &\textstyle\sum_{j=1}^{M} \omega_j\, f(u_h(t,x_{Kj})) \cdot \text{ grad } v_h(x_{Kj})|K|. \end{aligned} \tag{3.2.4}$$

Thus, we finally obtain the weak formulation:

$$\frac{d}{dt}\int_k u_h(t,x)v_h(x)dx + \sum_{e\in\partial K}\ \sum_{l=1}^{L}\omega_l\, h_{e,K}(t,x_{el})\, v(x_{el})|e|$$
$$-\sum_{j=1}^{M}\omega_j\, f(u_h(t,x_{Kj}))\cdot \text{grad}v_h(x_{Kj})|K| = 0, \quad \forall v_h\in V_h, \quad \forall K\in\mathbb{T}_h.$$

These equations can be rewritten in ODE form as $\frac{d}{dt}u_h = L_h(u_h,\gamma_h)$. This defines the operator $L_h(u_h)$, which is a discrete approximation of $-div\, f(u)$. The following result gives an indication of the quality of this approximation.

PROPOSITION 3.1. Let $f(u)\in W^{k+2,\infty}(\Omega)$, and set $\gamma = trace(u)$. Let the quadrature rule over the edges be exact for polynomials of degree $(2k+1)$, and let the one over the element be exact for polynomials of degree $(2k)$. Assume that the family of triangulations $\mathbb{F}=\{\mathbb{T}_h\}_{h>0}$ is regular, i.e., that there is a constant σ such that:

$$\frac{h_K}{\rho_K}\geq\sigma, \quad \forall K\in\mathbb{T}_h, \quad \forall\mathbb{T}_h\in\mathbb{F}, \tag{3.2.5}$$

where h_K is the diameter of K, and ρ_K is the diameter of the biggest ball included in K. Then, if $V(K)\supset P^k(K), \quad \forall\ K\in\mathbb{T}_h$:

$$\|L_h(u,\gamma)+div\, f(u)\|_{L^\infty(\Omega)}\leq C\ h^{k+1}|f(u)|_{W^{k+2,\infty}(\Omega)}.$$

For a proof, see [**13**].

3.2.2. The form of the generalized slope limiter $\Lambda\Pi_h$. The construction of generalized slope limiters $\Lambda\Pi_h$ for several space dimensions is not a trivial matter and will not be discussed in these notes; we refer the interested reader to the paper by Cockburn, Hou, and Shu [**13**].

In these notes, we restrict ourselves to displaying very simple, practical, and effective generalized slope limiters $\Lambda\Pi_h$ which are closely related to the generalized slope limiters $\Lambda\Pi_h^k$ of the previous section.

To compute $\Lambda\Pi_h u_h$, we rely on the *assumption* that spurious oscillations are present in u_h only if they are present in its P^1 part u_h^1, which is its L^2-projection into the space of piecewise linear functions V_h^1. Thus, if they are not present in u_h^1, i.e., if

$$u_h^1 = \Lambda\Pi_h\, u_h^1,$$

then we assume that they are not present in u_h and hence do not do any limiting:

$$\Lambda\Pi_h\, u_h\ =\ u_h\,.$$

On the other hand, if spurious oscillations are present in the P^1 part of the solution u_h^1, i.e., if

$$u_h^1\neq\Lambda\Pi_h\, u_h^1,$$

then we chop off the higher order part of the numerical solution, and limit the remaining P^1 part:

$$\Lambda\Pi_h\, u_h\ =\ \Lambda\Pi_h\, u_h^1.$$

In this way, in order to define $\Lambda\Pi_h$ for arbitrary space V_h, we only need to actually define it for piecewise linear functions V_h^1. The exact way to do that, both for the triangular elements and for the rectangular elements, will be discussed in the next section.

3.3. Algorithm and implementation details

In this section we give the algorithm and implementation details, including numerical fluxes, quadrature rules, degrees of freedom, fluxes, and limiters of the RKDG method for both piecewise-linear and piecewise-quadratic approximations in both triangular and rectangular elements.

3.3.1. Fluxes. The numerical flux we use is the simple Lax-Friedrichs flux:

$$h_{e,K}(a,b) = \frac{1}{2}\left[\mathbf{f}(a)\cdot n_{e,K} + \mathbf{f}(b)\cdot n_{e,K} - \alpha_{e,K}\,(b-a)\right].$$

The numerical viscosity constant $\alpha_{e,K}$ should be an estimate of the biggest eigenvalue of the Jacobian $\frac{\partial}{\partial u}\mathbf{f}(u_h(x,t))\cdot n_{e,K}$ for (x,t) in a neighborhood of the edge e.

For the triangular elements, we use the local Lax-Friedrichs recipe:

- Take $\alpha_{e,K}$ to be the larger one of the largest eigenvalue (in absolute value) of $\frac{\partial}{\partial u}\mathbf{f}(\bar{u}_K)\cdot n_{e,K}$ and that of $\frac{\partial}{\partial u}\mathbf{f}(\bar{u}_{K'})\cdot n_{e,K}$, where $\bar{u}_K$ and $\bar{u}_{K'}$ are the means of the numerical solution in the elements K and K' sharing the edge e.

For the rectangular elements, we use the local Lax-Friedrichs recipe :

- Take $\alpha_{e,K}$ to be the largest of the largest eigenvalue (in absolute value) of $\frac{\partial}{\partial u}\mathbf{f}(\bar{u}_{K''})\cdot n_{e,K}$, where $\bar{u}_{K''}$ is the mean of the numerical solution in the element K'', which runs over all elements on the same line (horizontally or vertically, depending on the direction of $n_{e,K}$) with K and K' sharing the edge e.

3.3.2. Quadrature rules. According to the analysis done in [**13**], the quadrature rules for the edges of the elements, (3.2.3), must be exact for polynomials of degree $2k+1$, and the quadrature rules for the interior of the elements, (3.2.4), must be exact for polynomials of degree $2k$, if P^k methods are used. Here we discuss the quadrature points used for P^1 and P^2 in the triangular and rectangular element cases.

3.3.3. The rectangular elements. For the edge integral, we use the following two point Gaussian rule

$$\int_{-1}^{1} g(x)dx \approx g\left(-\frac{1}{\sqrt{3}}\right) + g\left(\frac{1}{\sqrt{3}}\right), \tag{3.3.1}$$

for the P^1 case, and the following three point Gaussian rule

$$\int_{-1}^{1} g(x)dx \approx \frac{5}{9}\left[g\left(-\frac{3}{5}\right) + g\left(\frac{3}{5}\right)\right] + \frac{8}{9}\,g(0), \tag{3.3.2}$$

for the P^2 case, suitably scaled to the relevant intervals.

For the interior of the elements, we could use a tensor product of (3.3.1), with four quadrature points, for the P^1 case. But to save cost, we “recycle” the values of the fluxes at the element boundaries, and only add one new quadrature point in the middle of the element. Thus, to approximate the integral $\int_{-1}^{1}\int_{-1}^{1} g(x,y)dxdy$, we use the following quadrature rule:

$$\approx \frac{1}{4}\left[g\left(-1,\frac{1}{\sqrt{3}}\right)+g\left(-1,-\frac{1}{\sqrt{3}}\right)+g\left(-\frac{1}{\sqrt{3}},-1\right)+g\left(\frac{1}{\sqrt{3}},-1\right)\right.$$
$$\left.+g\left(1,-\frac{1}{\sqrt{3}}\right)+g\left(1,\frac{1}{\sqrt{3}}\right)+g\left(\frac{1}{\sqrt{3}},1\right)+g\left(-\frac{1}{\sqrt{3}},1\right)\right]+2\,g(0,0).$$

For the P^2 case, we use a tensor product of (3.3.2), with 9 quadrature points.

3.3.4. The triangular elements. For the edge integral, we use the same two point or three point Gaussian quadratures as in the rectangular case, (3.3.1) and (3.3.2), for the P^1 and P^2 cases, respectively.

For the interior integrals (3.2.4), we use the three mid-point rule

$$\int_K g(x,y)dxdy \approx \frac{|K|}{3}\sum_{i=1}^{3} g(m_i)\,,$$

where m_i are the mid-points of the edges, for the P^1 case. For the P^2 case, we use a seven-point quadrature rule which is exact for polynomials of degree 5 over triangles.

3.3.5. Basis and degrees of freedom. We emphasize that the choice of basis and degrees of freedom does not affect the algorithm, as it is completely determined by the choice of function space $V(h)$, the numerical fluxes, the quadrature rules, the slope limiting, and the time discretization. However, a suitable choice of basis and degrees of freedom may simplify the implementation and calculation.

3.3.6. The rectangular elements. For the P^1 case, we use the following expression for the approximate solution $u_h(x,y,t)$ inside the rectangular element $[x_{i-\frac{1}{2}},x_{i+\frac{1}{2}}]\times[y_{j-\frac{1}{2}},y_{j+\frac{1}{2}}]$:

$$u_h(x,y,t)=\bar{u}(t)+u_x(t)\phi_i(x)+u_y(t)\psi_j(y) \tag{3.3.3}$$

where

$$\phi_i(x)=\frac{x-x_i}{\Delta x_i/2},\qquad \psi_j(y)=\frac{y-y_j}{\Delta y_j/2}, \tag{3.3.4}$$

and

$$\Delta x_i = x_{i+\frac{1}{2}}-x_{i-\frac{1}{2}},\qquad \Delta y_j = y_{j+\frac{1}{2}}-y_{j-\frac{1}{2}}.$$

The degrees of freedoms, to be evolved in time, are then

$$\bar{u}(t),\ \ u_x(t),\ \ u_y(t).$$

Here we have omitted the subscripts ij these degrees of freedom should have, to indicate that they belong to the element ij which is $[x_{i-\frac{1}{2}},x_{i+\frac{1}{2}}]\times[y_{j-\frac{1}{2}},y_{j+\frac{1}{2}}]$.

Notice that the basis functions

$$1,\ \ \phi_i(x),\ \ \psi_j(y),$$

are orthogonal, hence the local mass matrix is diagonal:

$$M=\Delta x_i\Delta y_j\,diag\left(1,\frac{1}{3},\frac{1}{3}\right).$$

For the P^2 case, the expression for the approximate solution $u_h(x,y,t)$ inside the rectangular element $[x_{i-\frac{1}{2}}, x_{i+\frac{1}{2}}] \times [y_{j-\frac{1}{2}}, y_{j+\frac{1}{2}}]$ is:

$$\begin{aligned} u_h(x,y,t) &= \bar{u}(t) + u_x(t)\phi_i(x) + u_y(t)\psi_j(y) + u_{xy}(t)\phi_i(x)\psi_j(y) \\ &\quad + u_{xx}(t)\left(\phi_i^2(x) - \frac{1}{3}\right) + u_{yy}(t)\left(\psi_j^2(y) - \frac{1}{3}\right), \end{aligned} \tag{3.3.5}$$

where $\phi_i(x)$ and $\psi_j(y)$ are defined by (3.3.4). The degrees of freedoms, to be evolved in time, are

$$\bar{u}(t),\ u_x(t),\ u_y(t),\ u_{xy}(t),\ u_{xx}(t),\ u_{yy}(t).$$

Again the basis functions

$$1,\ \phi_i(x),\ \psi_j(y),\ \phi_i(x)\psi_j(y),\ \phi_i^2(x) - \frac{1}{3},\ \psi_j^2(y) - \frac{1}{3},$$

are orthogonal, hence the local mass matrix is diagonal:

$$M = \Delta x_i \Delta y_j \, diag\left(1, \frac{1}{3}, \frac{1}{3}, \frac{1}{9}, \frac{4}{45}, \frac{4}{45}\right).$$

3.3.7. The triangular elements. For the P^1 case, we use the following expression for the approximate solution $u_h(x,y,t)$ inside the triangle K:

$$u_h(x,y,t) = \sum_{i=1}^{3} u_i(t)\varphi_i(x,y)$$

where the degrees of freedom $u_i(t)$ are values of the numerical solution at the midpoints of edges, and the basis function $\varphi_i(x,y)$ is the linear function which takes the value 1 at the mid-point m_i of the i-th edge, and the value 0 at the mid-points of the two other edges. The mass matrix is diagonal

$$M = |K| diag\left(\frac{1}{3}, \frac{1}{3}, \frac{1}{3}\right).$$

For the P^2 case, we use the following expression for the approximate solution $u_h(x,y,t)$ inside the triangle K:

$$u_h(x,y,t) = \sum_{i=1}^{6} u_i(t)\xi_i(x,y)$$

where the degrees of freedom, $u_i(t)$, are values of the numerical solution at the three midpoints of edges and the three vertices. The basis function $\xi_i(x,y)$, is the quadratic function which takes the value 1 at the point i of the six points mentioned above (the three midpoints of edges and the three vertices), and the value 0 at the remaining five points. The mass matrix this time is not diagonal.

3.3.8. Limiting. We construct slope limiting operators $\Lambda\Pi_h$ on piecewise linear functions u_h in such a way that the following properties are satisfied:

1. Accuracy: if u_h is linear then $\Lambda\Pi_h\, u_h = u_h$.
2. Conservation of mass: for every element K of the triangulation $\mathbb{T}_h$, we have:

$$\int_K \Lambda\Pi_h\, u_h = \int_K u_h.$$

3. Slope limiting: on each element K of $\mathbb{T}_h$, the gradient of $\Lambda\Pi_h\, u_h$ is not bigger than that of u_h.

The actual form of the slope limiting operators is closely related to that of the slope limiting operators studied in [**15**] and [**13**].

3.3.9. The rectangular elements. The limiting is performed on u_x and u_y in (3.3.3), using the differences of the means. For a scalar equation, u_x would be limited (replaced) by

$$\bar{m}\left(u_x, \bar{u}_{i+1,j} - \bar{u}_{ij}, \bar{u}_{ij} - \bar{u}_{i-1,j}\right) \tag{3.3.6}$$

where the function $\bar{m}$ is the TVB corrected *minmod* function defined in the previous section.

The TVB correction is needed to avoid unnecessary limiting near smooth extrema, where the quantity u_x or u_y is on the order of $O(\Delta x^2)$ or $O(\Delta y^2)$. For an estimate of the TVB constant M in terms of the second derivatives of the function, see [**15**]. Usually, the numerical results are not sensitive to the choice of M in a large range. In all the calculations in this paper we take M to be 50.

Similarly, u_y is limited (replaced) by

$$\bar{m}(u_y, \bar{u}_{i,j+1} - \bar{u}_{ij}, \bar{u}_{ij} - \bar{u}_{i,j-1}).$$

with a change of Δx to Δy in (3.3.6).

For systems, we perform the limiting in the local characteristic variables. To limit the vector u_x in the element ij, we proceed as follows:

- Find the matrix R and its inverse R^{-1}, which diagonalize the Jacobian evaluated at the mean in the element ij in the x-direction:

$$R^{-1}\,\frac{\partial f_1(\bar{u}_{ij})}{\partial u} R = \Lambda\,,$$

  where Λ is a diagonal matrix containing the eigenvalues of the Jacobian. Notice that the columns of R are the right eigenvectors of $\frac{\partial f_1(\bar{u}_{ij})}{\partial u}$ and the rows of R^{-1} are the left eigenvectors.
- Transform all quantities needed for limiting, i.e., the three vectors u_{xij}, $\bar{u}_{i+1,j} - \bar{u}_{ij}$ and $\bar{u}_{ij} - \bar{u}_{i-1,j}$, to the characteristic fields. This is achieved by left multiplying these three vectors by R^{-1}.
- Apply the scalar limiter (3.3.6) to each of the components of the transformed vectors.
- The result is transformed back to the original space by left multiplying R on the left.

3.3.10. The triangular elements. To construct the slope limiting operators for triangular elements, we proceed as follows. We start by making a simple observation. Consider the triangles in Figure 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 we have that

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

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

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

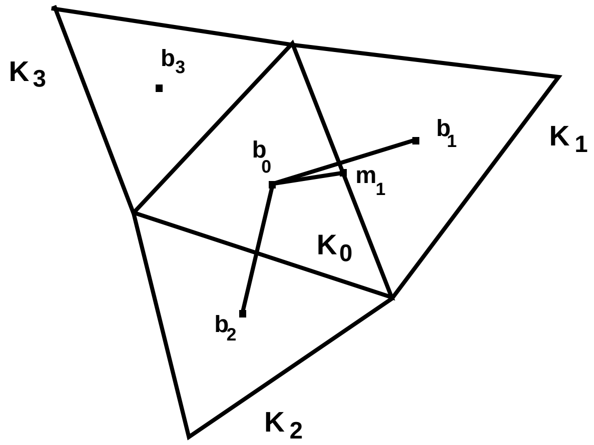


FIGURE 1. Illustration of limiting.

and since

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

we have that

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

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

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

To compute $\Lambda\Pi_h u_h$, we first compute the quantities

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

where $\bar{m}$ is the TVB modified *minmod* function and $\nu > 1$. We take $\nu = 1.5$ in our numerical runs. Then, if $\sum_{i=1}^{3}\Delta_i = 0$, we simply set

$$\Lambda\Pi_h u_h(x, y) = \bar{u}_{K_0} + \sum_{i=1}^{3} \Delta_i\,\varphi_i(x, y).$$

If $\sum_{i=1}^{3}\Delta_i \neq 0$, we compute

$$pos = \sum_{i=1}^{3} \max(0, \Delta_i), \qquad neg = \sum_{i=1}^{3} \max(0, -\Delta_i),$$

and set

$$\theta^+ = \min\left(1, \frac{neg}{pos}\right), \qquad \theta^- = \min\left(1, \frac{pos}{neg}\right).$$

Then, we define

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

where

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

It is very easy to see that this slope limiting operator satisfies the three properties listed above.

For systems, we perform the limiting in the local characteristic variables. To limit Δ_i, we proceed as in the rectangular case, the only difference being that we work with the following Jacobian

$$\frac{\partial}{\partial u} f(\bar{u}_{K_0}) \cdot \frac{m_i - b_0}{|m_i - b_0|} \; .$$

3.4. Computational results: Transient, nonsmooth solutions

In this section we present several numerical results obtained with the P^1 and P^2 (second and third order accurate) RKDG methods with either rectangles or triangles in the triangulation. These are standard test problems for Euler equations of compressible gas dynamics.

3.4.1. The double-Mach reflection problem. Double Mach reflection of a strong shock. This problem was studied extensively in Woodward and Colella [**66**] and later by many others. We use exactly the same setup as in [**66**], namely a Mach 10 shock initially makes a 60° angle with a reflecting wall. The undisturbed air ahead of the shock has a density of 1.4 and a pressure of 1.

For the rectangle based triangulation, we use a rectangular computational domain $[0,4]\times[0,1]$, as in [**66**]. The reflecting wall lies at the bottom of the computational domain for $\frac{1}{6} \le x \le 4$. Initially a right-moving Mach 10 shock is positioned at $x = \frac{1}{6}, y = 0$ and makes a 60° angle with the x-axis. For the bottom boundary, the exact post-shock condition is imposed for the part from $x = 0$ to $x = \frac{1}{6}$, to mimic an angled wedge. Reflective boundary condition is used for the rest. At the top boundary of our computational domain, the flow values are set to describe the exact motion of the Mach 10 shock. Inflow/outflow boundary conditions are used for the left and right boundaries. As in [**66**], only the results in $[0,3]\times[0,1]$ are displayed.

For the triangle based triangulation, we have the freedom to treat irregular domains and thus use a true wedged computational domain. Reflective boundary conditions are then used for all the bottom boundary, including the sloped portion. Other boundary conditions are the same as in the rectangle case.

Uniform rectangles are used in the rectangle based triangulations. Four different meshes are used: 240×60 rectangles ($\Delta x = \Delta y = \frac{1}{60}$); 480×120 rectangles ($\Delta x = \Delta y = \frac{1}{120}$); 960×240 rectangles ($\Delta x = \Delta y = \frac{1}{240}$); and 1920×480 rectangles ($\Delta x = \Delta y = \frac{1}{480}$). The density is plotted in Figure 2 for the P^1 case and in 3 for the P^2 case.

To better appreciate the difference between the P^1 and P^2 results in these pictures, we show a "blowed up" portion around the double Mach region in Figure 4 and show one-dimensional cuts along the line $y = 0.4$ in Figures 5 and 6. In Figure 4, w 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.

Moreover, from Figure 5, we clearly see that the difference between the results obtained by using P^1 and P^2, on the same mesh, increases dramatically as the mesh size decreases. This indicates that the use of polynomials of high degree might be beneficial for capturing the above mentioned structures. From Figure 6, we see that the results obtained with P^1 are qualitatively similar to those obtained with P^2 in a coarser mesh; the similarity increases as the meshsize decreases. 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 of the 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.

The optimal strategy, of course, is to use adaptivity and concentrate triangles around the interesting region, and/or change the order of the scheme in different regions.

3.4.2. The forward-facing step problem. Flow past a forward facing step. This problem was again studied extensively in Woodward and Colella [**66**] and later by many others. The set up of the problem is the following: A right going Mach 3 uniform flow enters a wind tunnel of 1 unit wide and 3 units long. The step is 0.2 units high and is located 0.6 units from the left-hand end of the tunnel. The problem is initialized by a uniform, right-going Mach 3 flow. Reflective boundary conditions are applied along the walls of the tunnel and in-flow and out-flow boundary conditions are applied at the entrance (left-hand end) and the exit (right-hand end), respectively.

The corner of the step is a singularity, which we study carefully in our numerical experiments. Unlike in [**66**] and many other papers, we do not modify our scheme near the corner in any way. It is well known that this leads to an errorneous entropy layer at the downstream bottom wall, as well as a spurious Mach stem at the bottom wall. However, these artifacts decrease when the mesh is refined. In Figure 7, second order P^1 results using rectangle triangulations 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 increased mesh points. In Figure 8, third order P^2 results using the same meshes are shown.

In order to verify that the erroneous entropy layer at the downstream bottom wall and the spurious Mach stem at the bottom wall are both artifacts caused by the corner singularity, we use our triangle code to locally refine near the corner progressively; we use the meshes displayed in Figure 9. In Figure 10, we plot the density obtained by the P^1 triangle code, with triangles (roughly the resolution of $\Delta x = \Delta y = \frac{1}{40}$, except around the corner). In Figure 11, we plot the entropy around the corner for the same runs. We can see that, with more triangles concentrated near the corner, the artifacts gradually decrease. Results with P^2 codes in Figures 12 and 13 show a similar trend.

3.5. Computational results: Steady state, smooth solutions

In this section, we present some of the numerical results of Bassi and Rebay [**2**] in two dimensions and Warburton, Lomtev, Kirby and Karniadakis [**65**] in three dimensions.

The purpose of the numerical results of Bassi and Rebay [**2**] we are presenting is to assess (i) the effect of the quality of the approximation of curved boundaries and of (ii) the effect of the degree of the polynomials on the quality of the approximate solution. The test problem we consider here is the two-dimensional steady-state, subsonic flow around a disk at Mach number $M_\infty = 0.38$. Since the solution is smooth and can be computed analytically, the quality of the approximation can be easily assessed.

In the figures 14, 15, 16, and 17, details of the meshes around the disk are shown together with the approximate solution given by the RKDG method using piecewise linear elements. These meshes approximate the circle with a polygonal. It can be seen that the approximate solution are of very low quality even for the most refined grid. This is an effect caused by the kinks of the polygonal approximating the circle.

This statement can be easily verified by taking a look to the figures 18, 19, 20, and 21. In these pictures the approximate solutions with piecewise linear, quadratic, and cubic elements are shown; the meshes have been modified to render *exactly* the circle. It is clear that the improvement in the quality of the approximation is enormous. Thus, a high-quality approximation of the boundaries has a dramatic improvement on the quality of the approximations.

Also, it can be seen that the higher the degree of the polynomials, the better the quality of the approximations, in particular from figures 18 and 19. In [**2**], Bassi and Rebay show that the RKDG method using polynomilas of degree k are $(k+1)$-th order accurate for $k = 1, 2, 3$. As a consequence, a RKDG method using polynomials of a higher degree is more efficient than a RKDG method using polynomials of lower degree.

In [**65**], Warburton, Lomtev, Kirby and Karniadakis present the same test problem in a three dimensional setting. In Figure 22, we can see the three-dimensional mesh and the density isosurfaces. We can also see how, while the mesh is being kept fixed and the degree of the polynomials k is increased from 1 to 9, the maximum error on the entropy goes exponentialy to zero. (In the picture, a so-called 'mode' is equal to $k+1$).

3.6. Concluding remarks

In this section, we have extended the RKDG methods to multidimensional systems. We have described in full detail the algorithms and displayed numerical results showing 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.

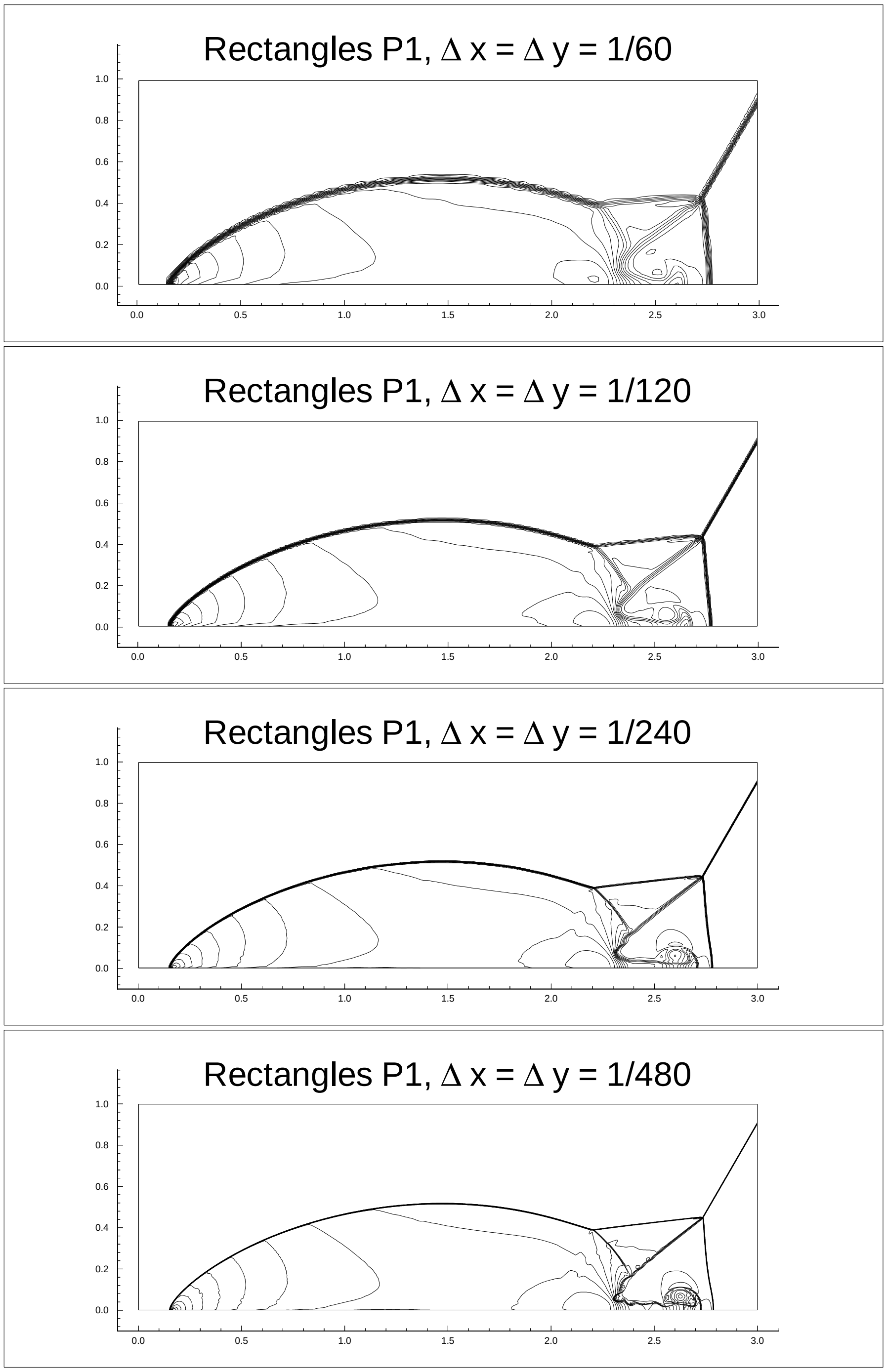


FIGURE 2. Double Mach reflection problem. Second order P^1 results. Density ρ. 30 equally spaced contour lines from $\rho = 1.3965$ to $\rho = 22.682$. Mesh refinement study. From top to bottom: $\Delta x = \Delta y = \frac{1}{60}, \frac{1}{120}, \frac{1}{240}$, and $\frac{1}{480}$.

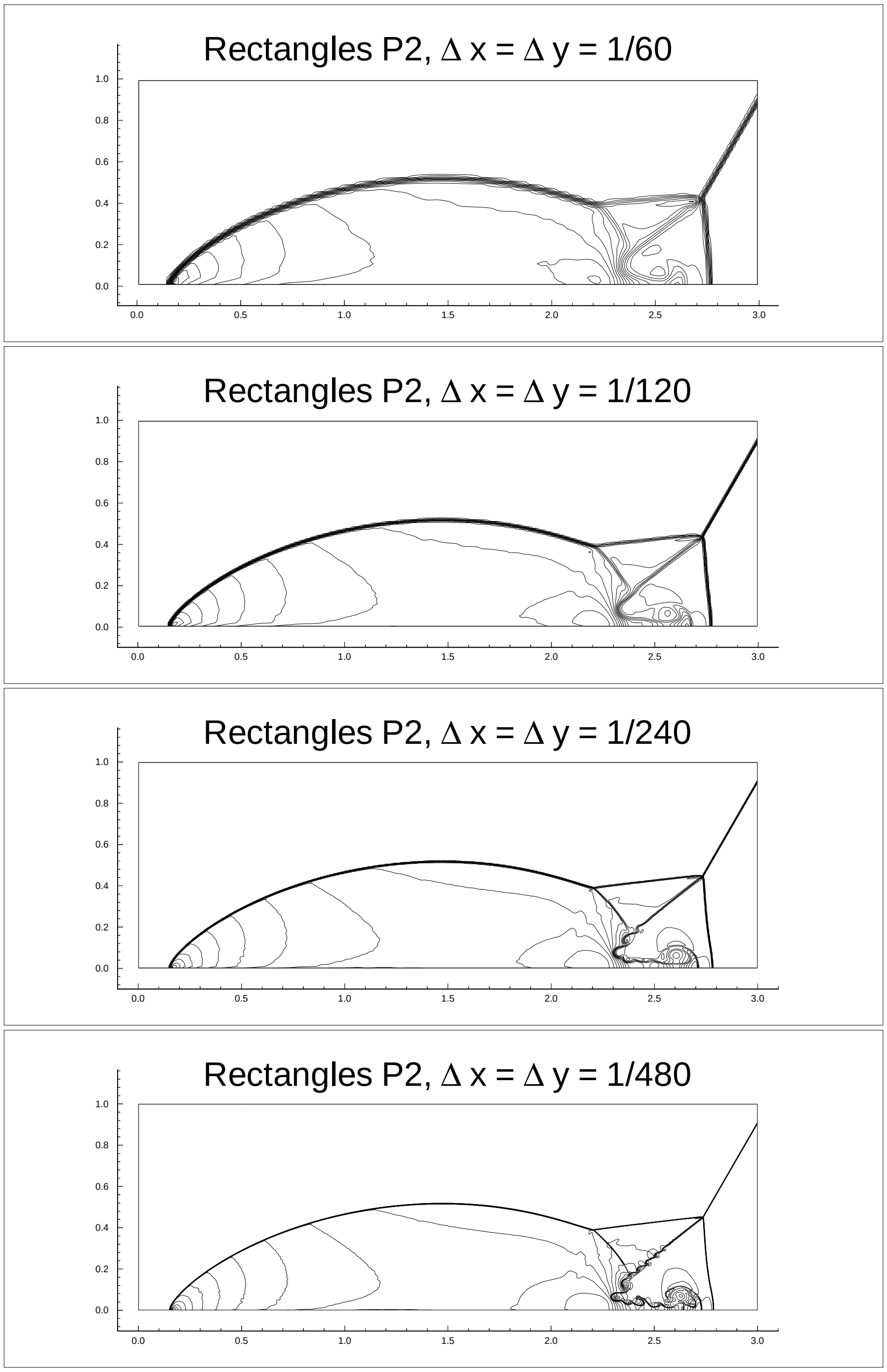


FIGURE 3. Double Mach reflection problem. Third order P^2 results. Density ρ. 30 equally spaced contour lines from $\rho = 1.3965$ to $\rho = 22.682$. Mesh refinement study. From top to bottom: $\Delta x = \Delta y = \frac{1}{60}, \frac{1}{120}, \frac{1}{240}$, and $\frac{1}{480}$.

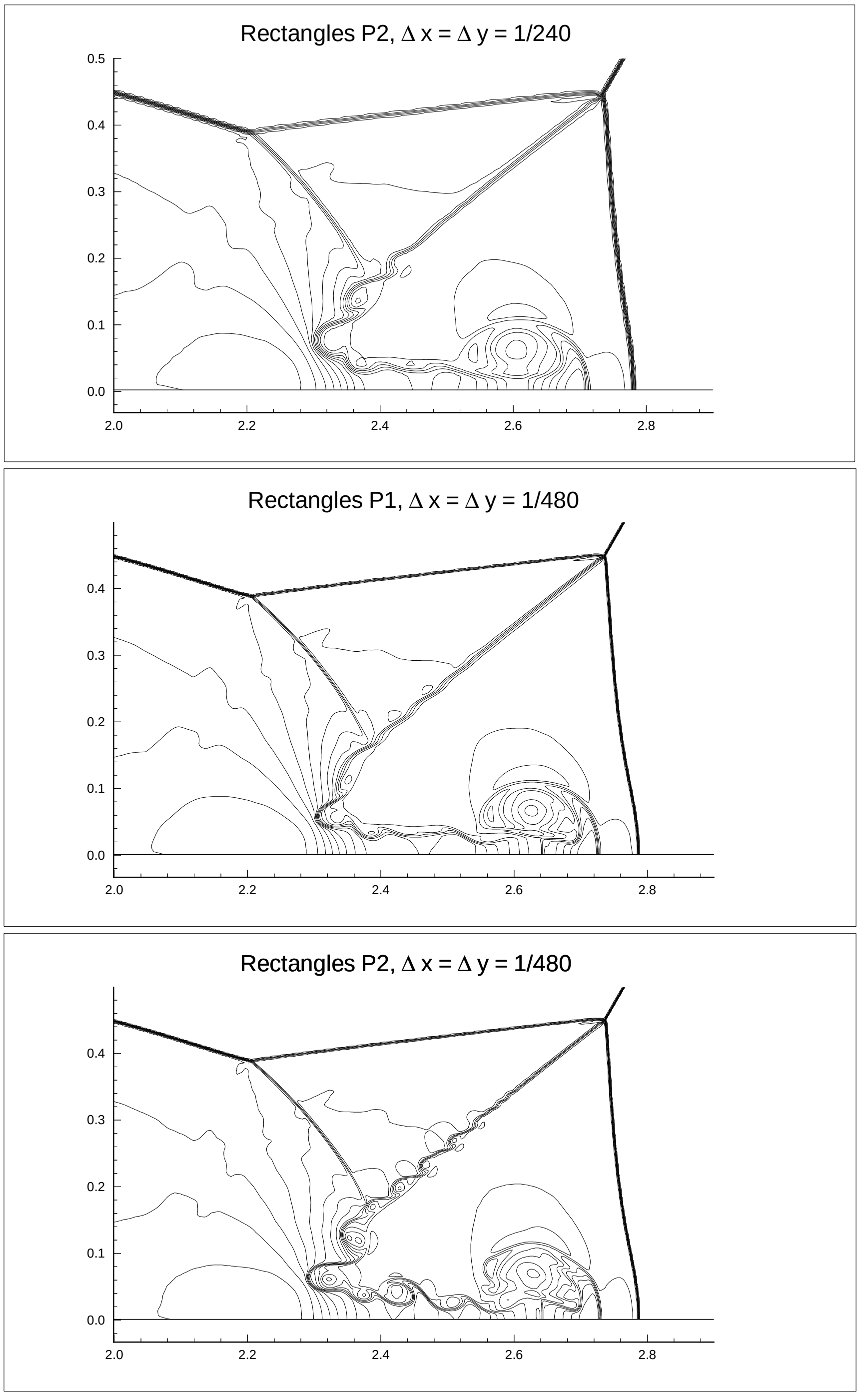


FIGURE 4. Double Mach reflection problem. Blowed-up region around the double Mach stems. Density ρ. Third order P^2 with $\Delta x = \Delta y = \frac{1}{240}$ (top); second order P^1 with $\Delta x = \Delta y = \frac{1}{480}$ (middle); and third order P^2 with $\Delta x = \Delta y = \frac{1}{480}$ (bottom).

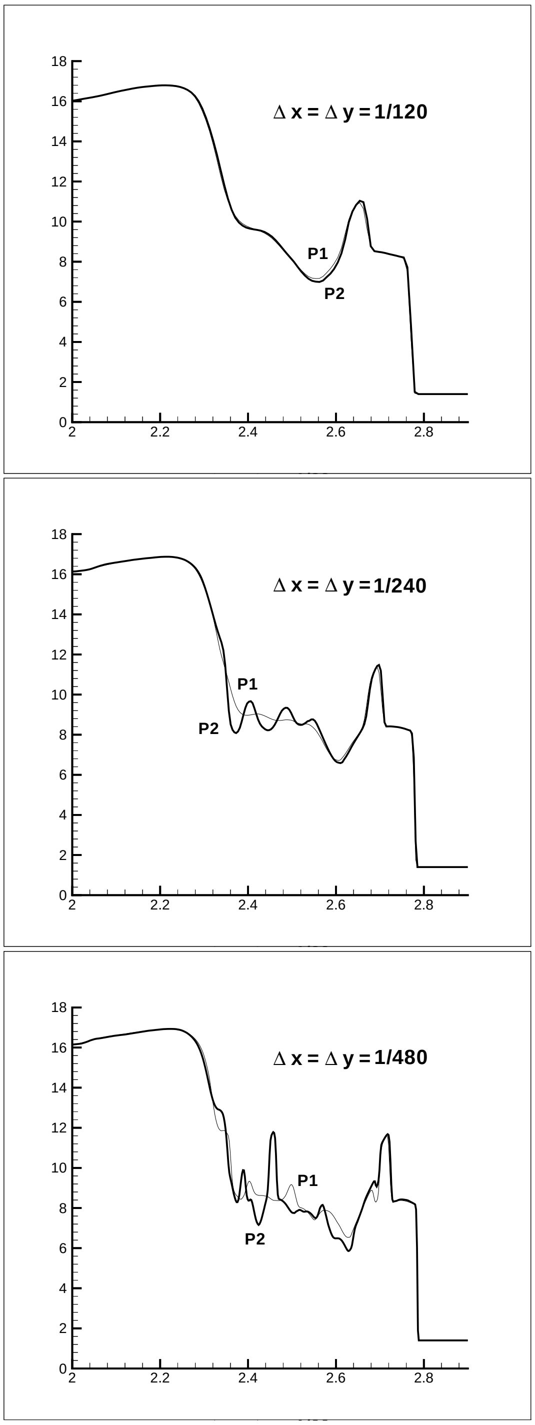


FIGURE 5. Double Mach reflection problem. Cut $y = 0.4$ of the blowed-up region. Density ρ. Comparison of second order P^1 with third order P^2 on the same mesh

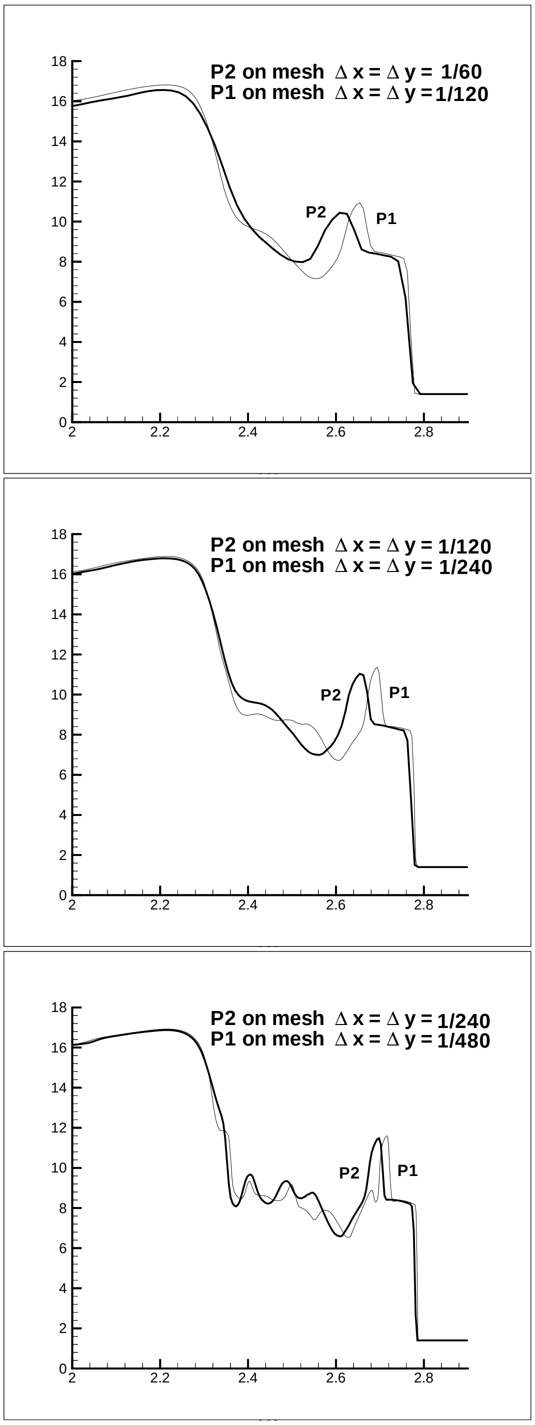


FIGURE 6. Double Mach reflection problem. Cut $y = 0.4$ of the blowed-up region. Density ρ. Comparison of second order P^1 with third order P^2 on a coarser mesh

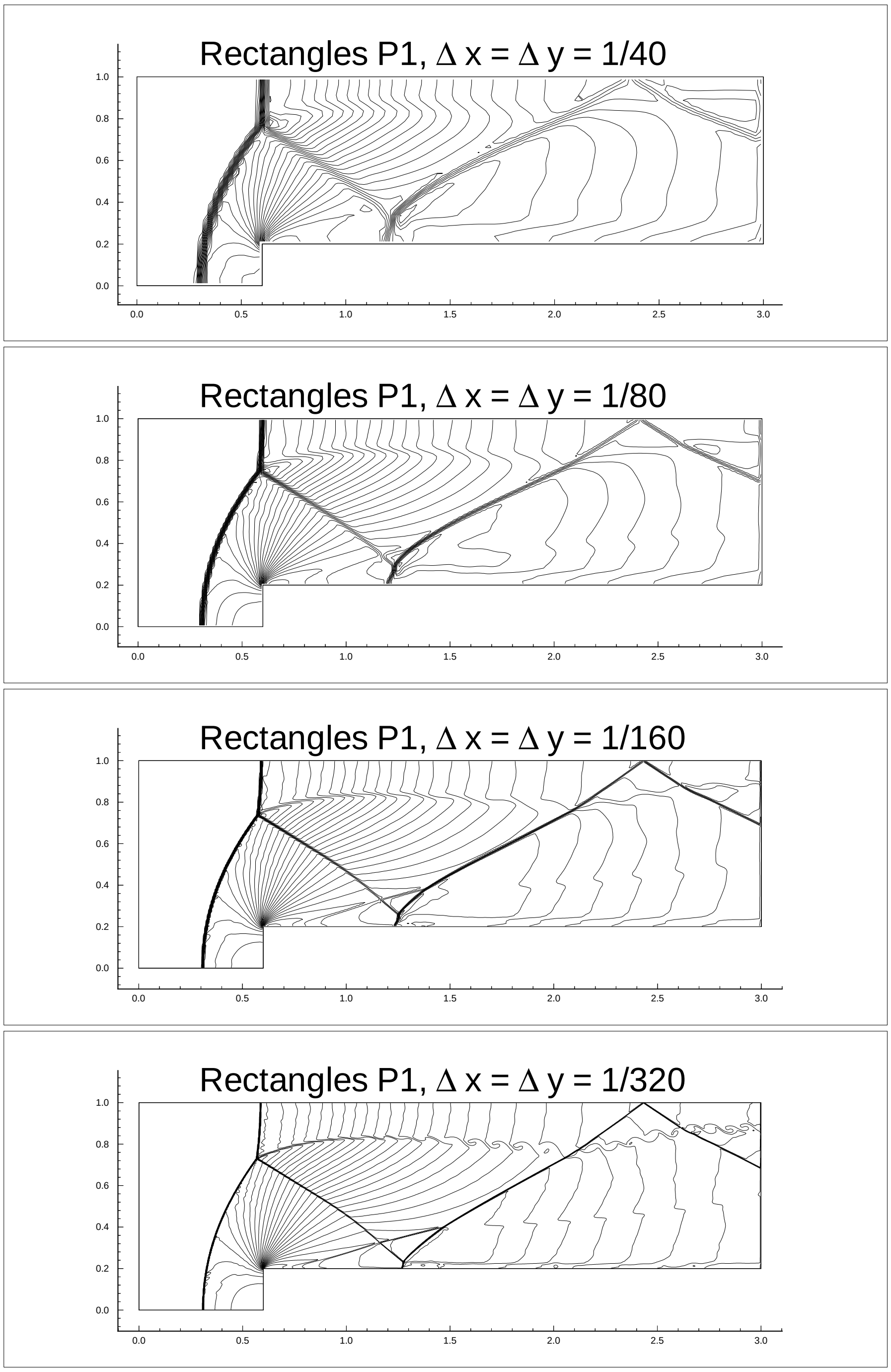


FIGURE 7. Forward facing step problem. Second order P^1 results. Density ρ. 30 equally spaced contour lines from $\rho = 0.090338$ to $\rho = 6.2365$. Mesh refinement study. From top to bottom: $\Delta x = \Delta y = \frac{1}{40}, \frac{1}{80}, \frac{1}{160}$, and $\frac{1}{320}$.

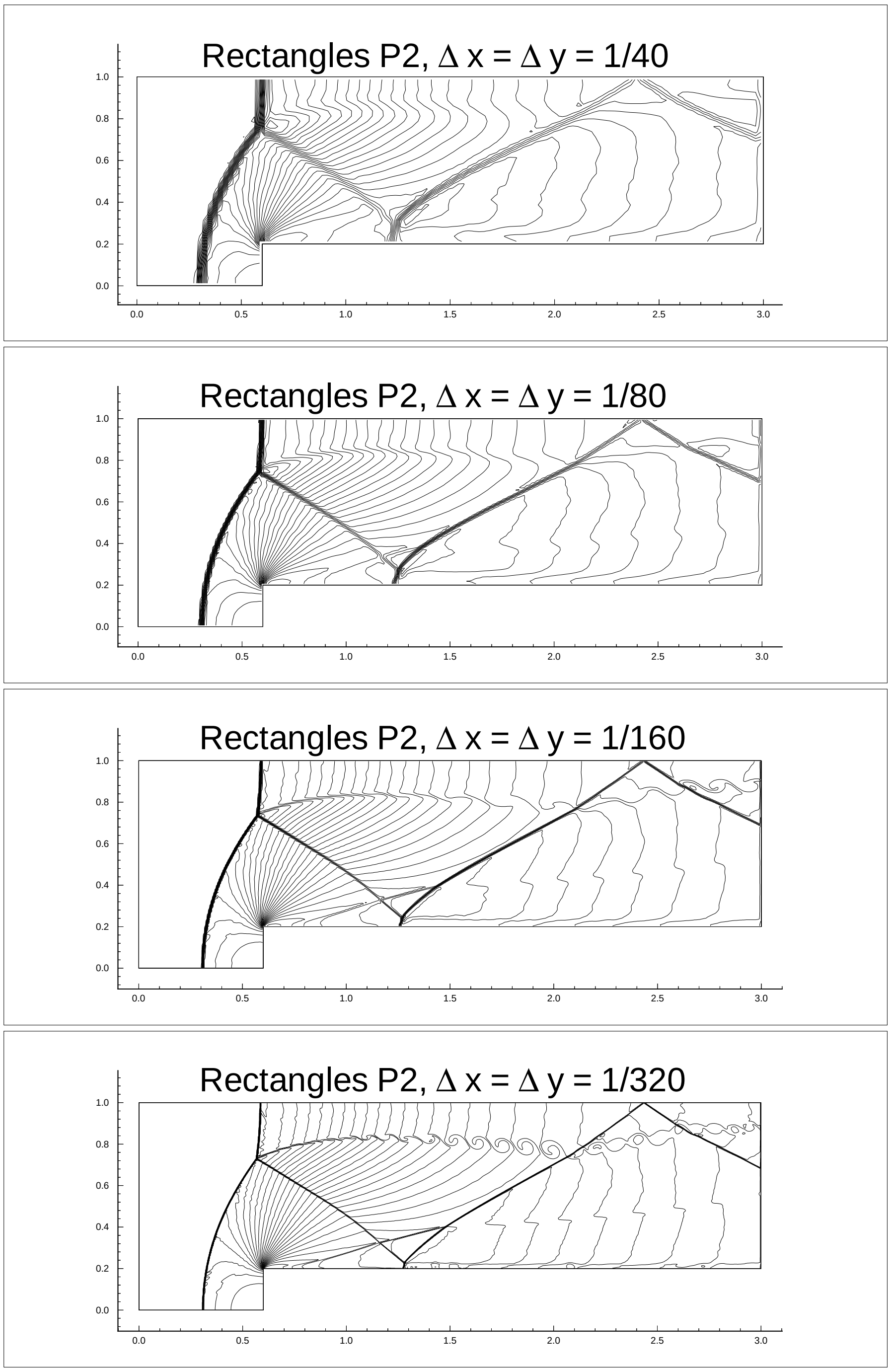


FIGURE 8. Forward facing step problem. Third order P^2 results. Density ρ. 30 equally spaced contour lines from $\rho = 0.090338$ to $\rho = 6.2365$. Mesh refinement study. From top to bottom: $\Delta x = \Delta y = \frac{1}{40}, \frac{1}{80}, \frac{1}{160}$, and $\frac{1}{320}$.

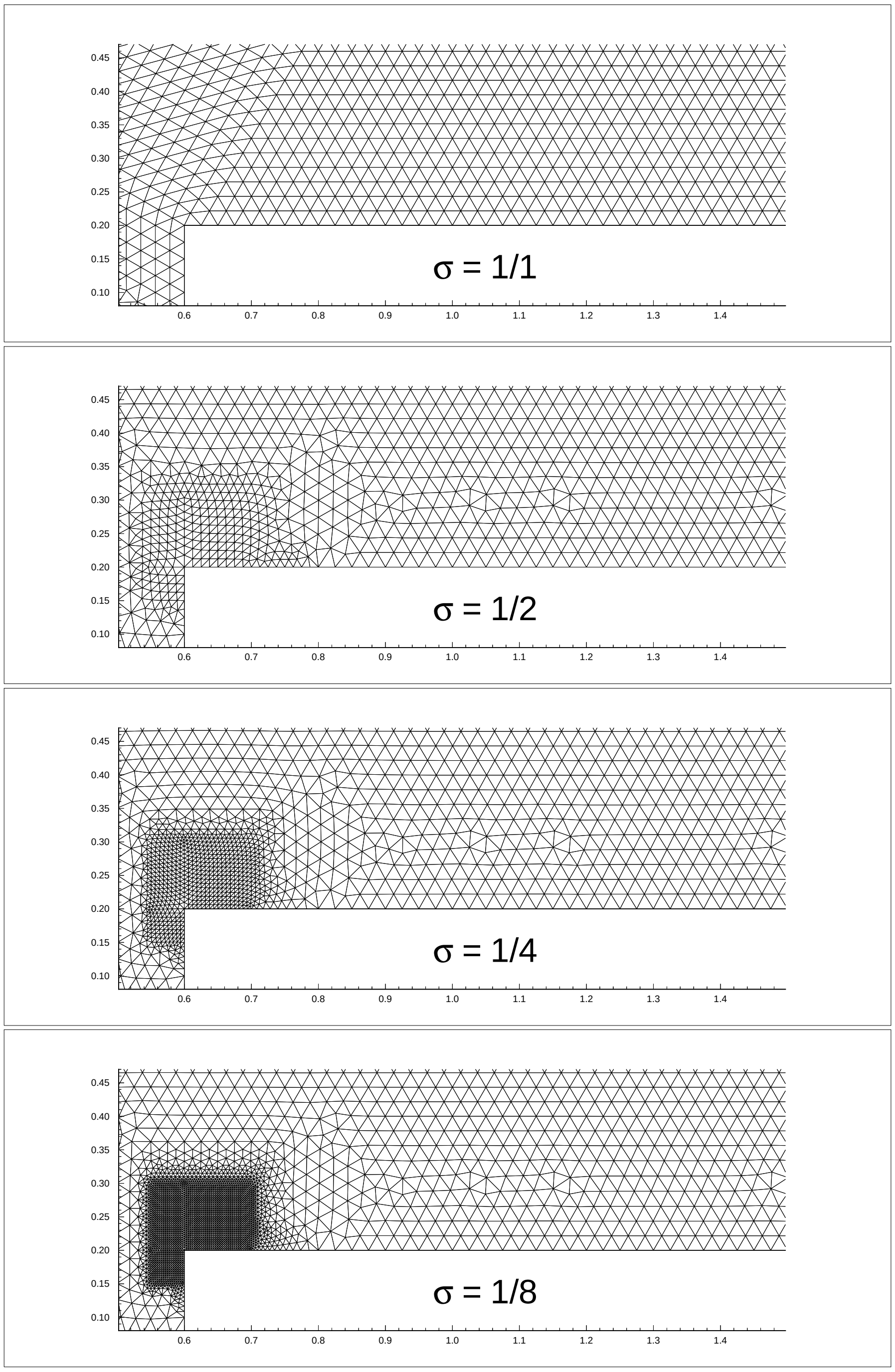


FIGURE 9. Forward facing step problem. Detail of the triangulations associated with the different values of σ. The parameter σ is the ratio between the typical size of the triangles near the corner and that elsewhere.

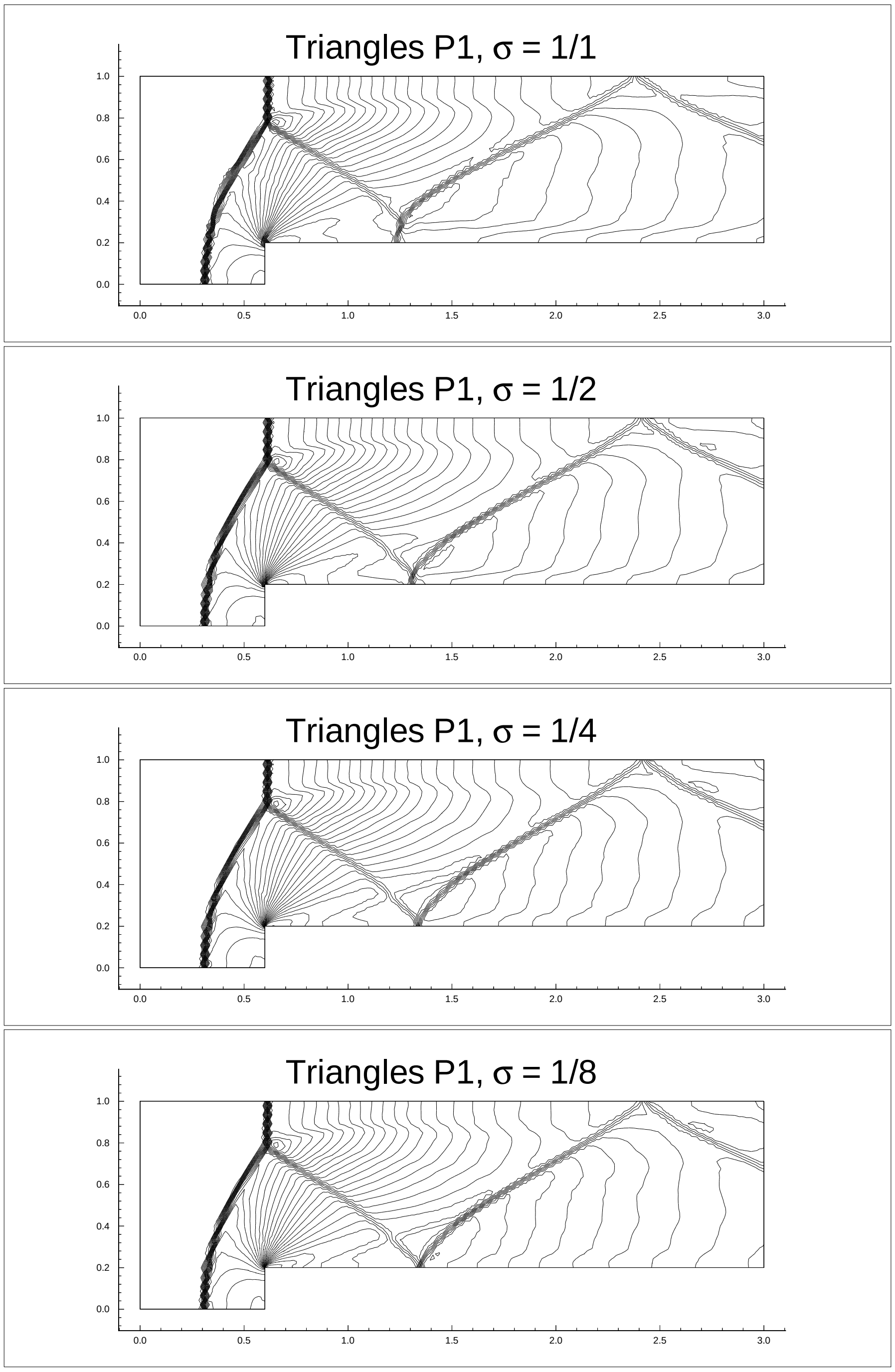


FIGURE 10. Forward facing step problem. Second order P^1 results. Density ρ. 30 equally spaced contour lines from $\rho = 0.090338$ to $\rho = 6.2365$. Triangle code. Progressive refinement near the corner

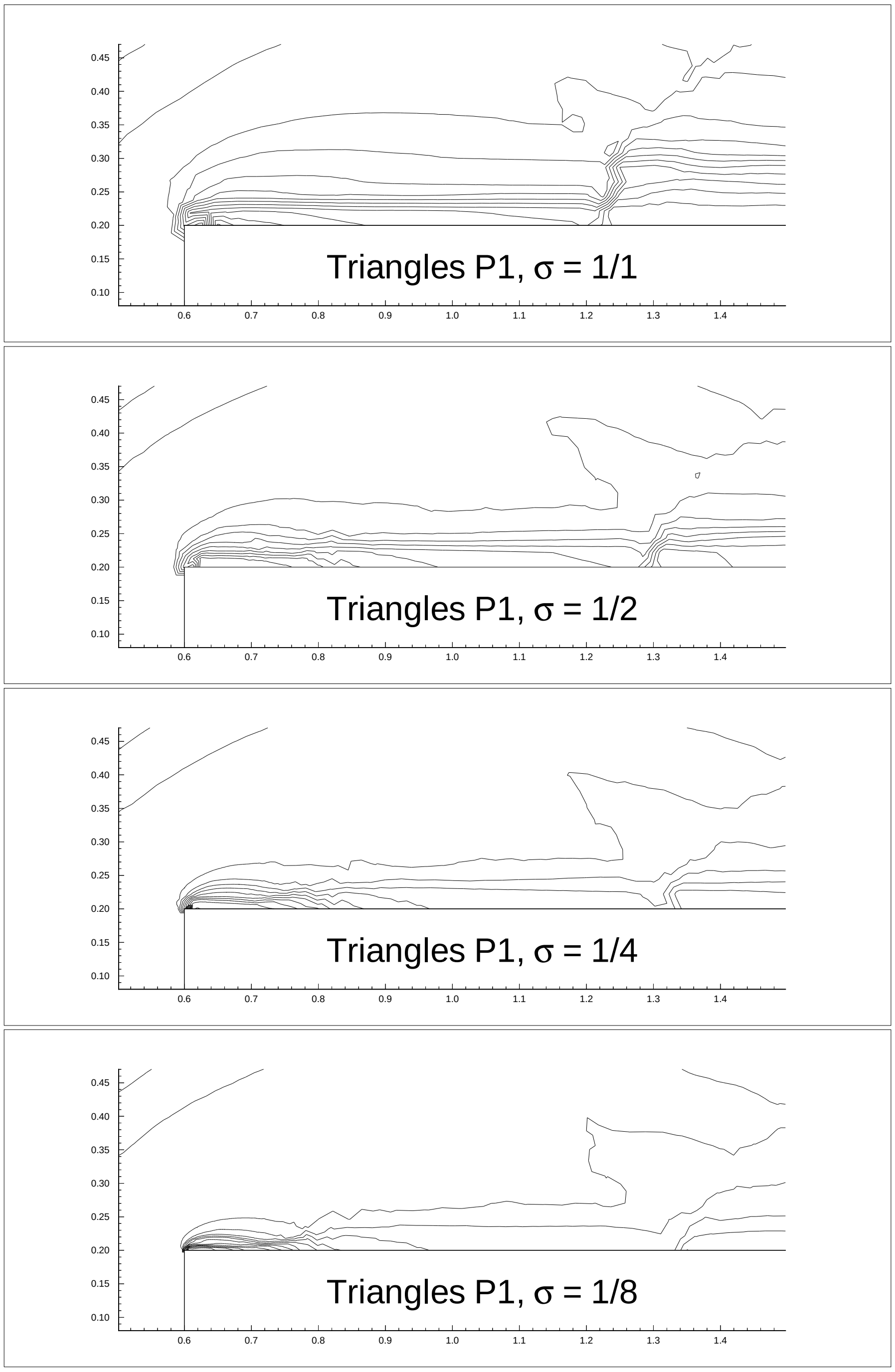


FIGURE 11. Forward facing step problem. Second order P^1 results. Entropy level curves around the corner. Triangle code. Progressive refinement near the corner

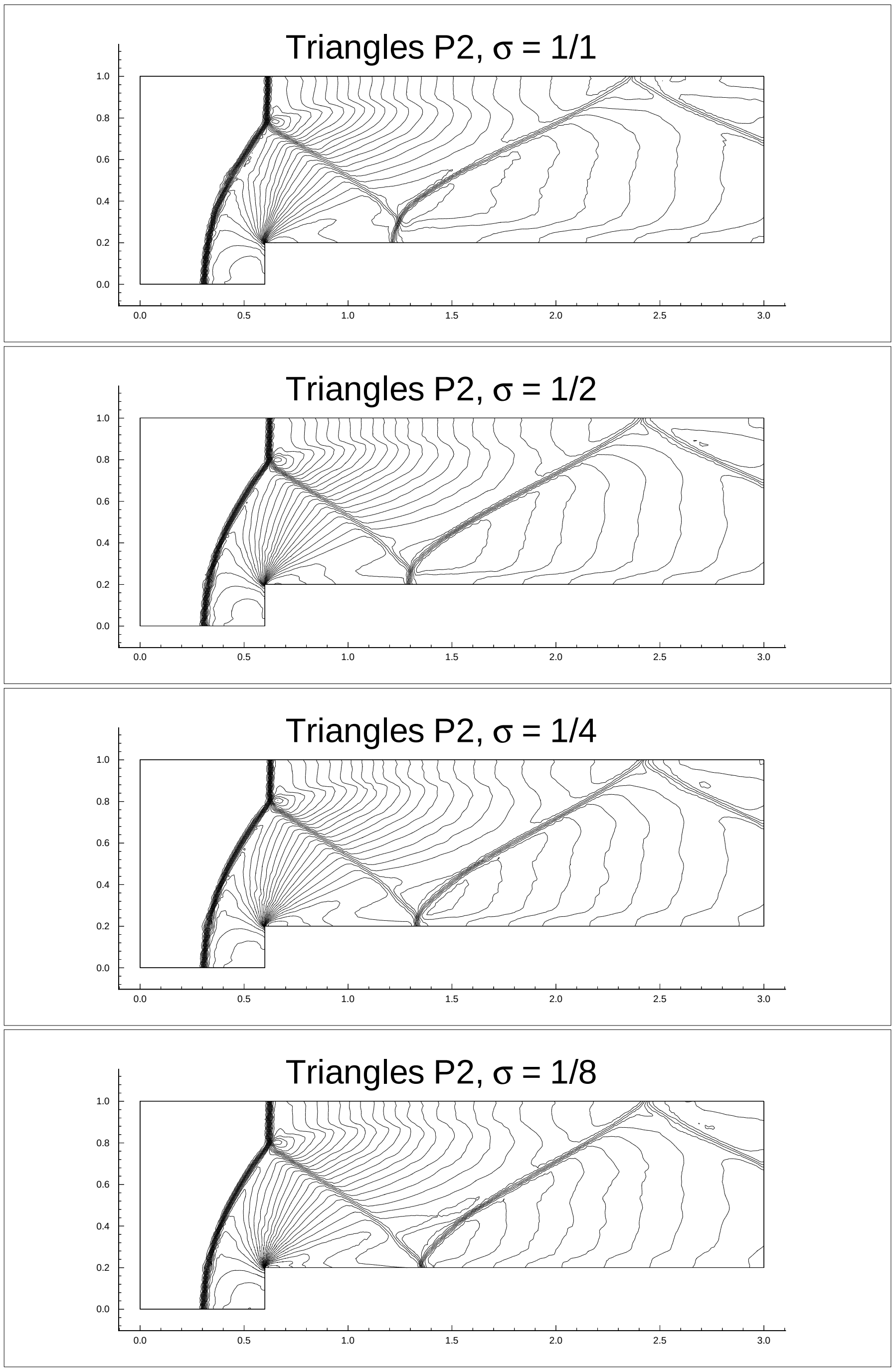


FIGURE 12. Forward facing step problem. Third order P^2 results. Density ρ. 30 equally spaced contour lines from $\rho = 0.090338$ to $\rho = 6.2365$. Triangle code. Progressive refinement near the corner

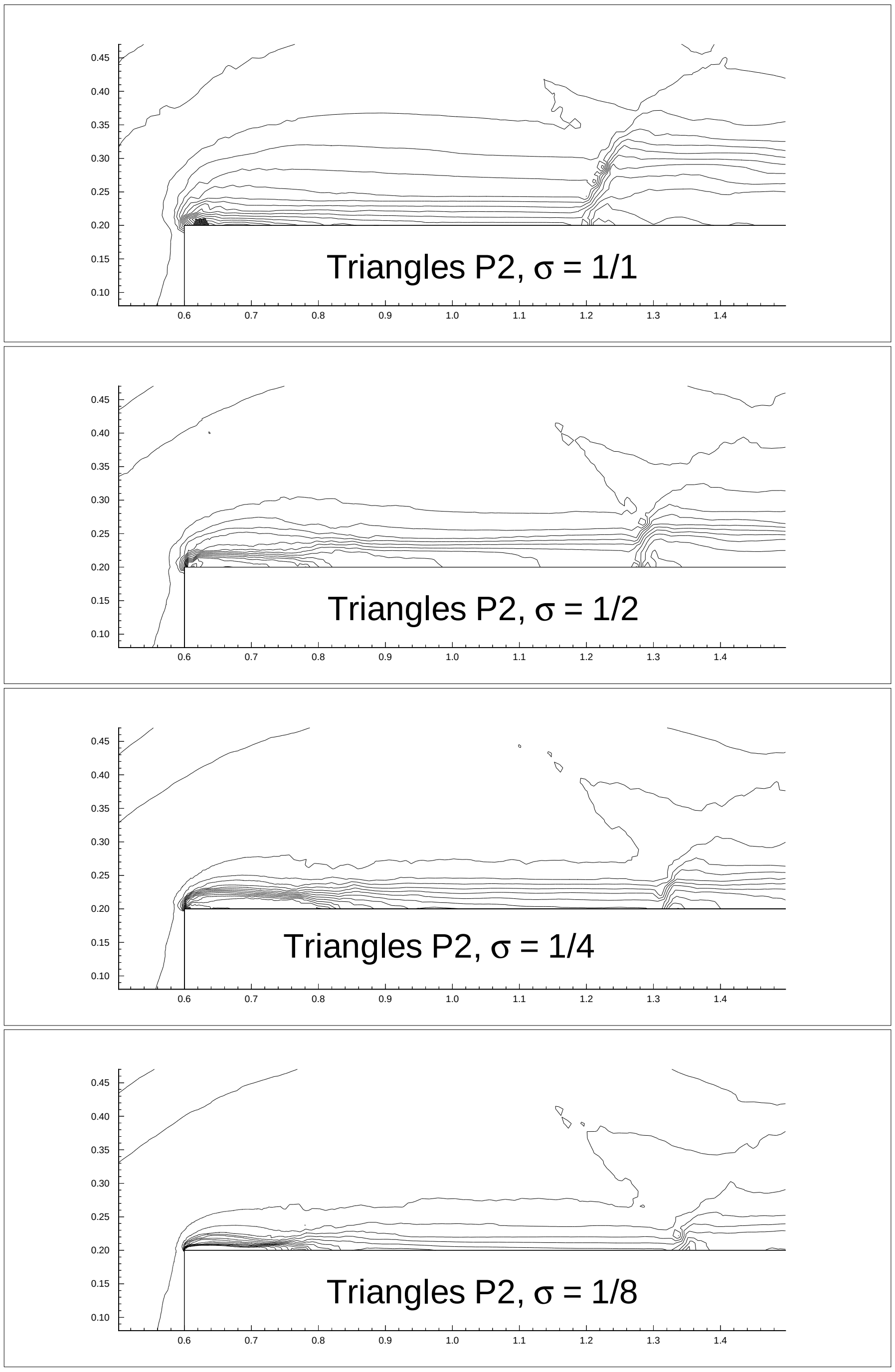


FIGURE 13. Forward facing step problem. Third order P^1 results. Entropy level curves around the corner. Triangle code. Progressive refinement near the corner

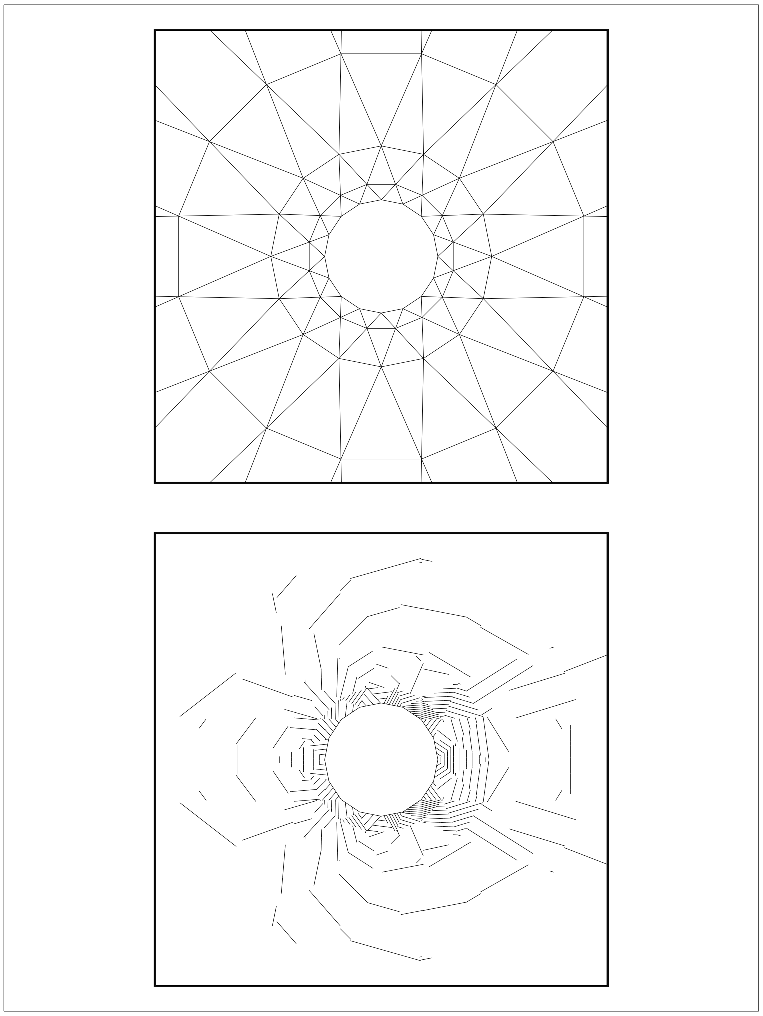


FIGURE 14. Grid “16 × 8” with a piecewise linear approximation of the circle (top) and the corresponding solution (Mach isolines) using P^1 elements (bottom).

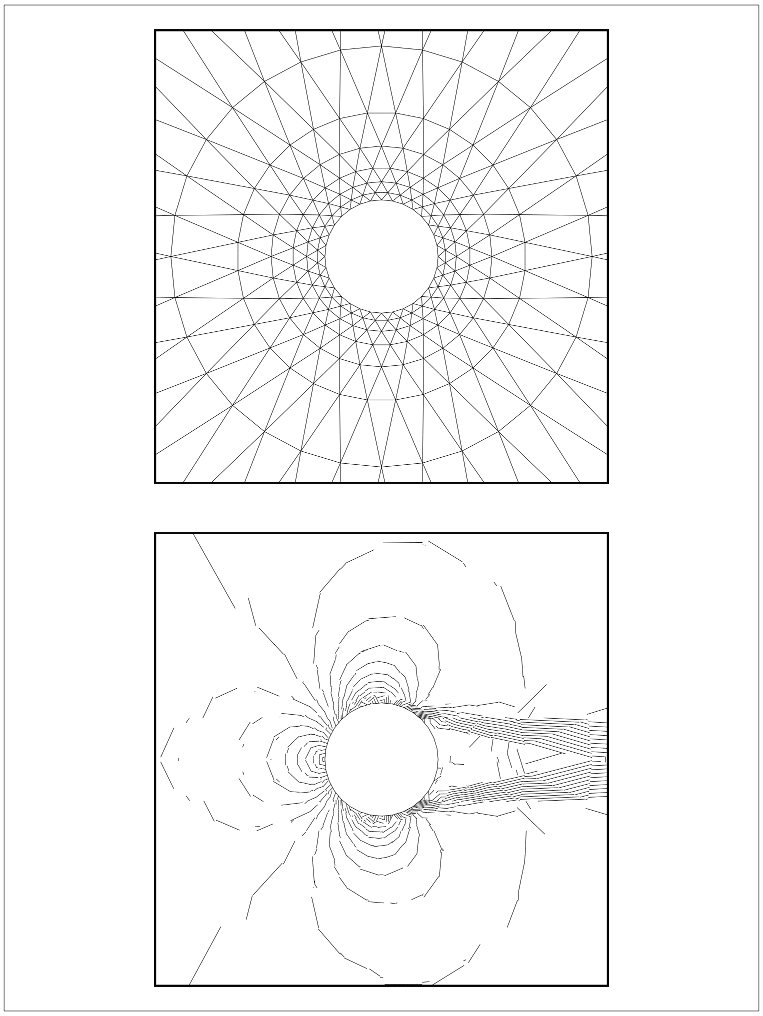


FIGURE 15. Grid "32×8" with a piecewise linear approximation of the circle (top) and the corresponding solution (Mach isolines) using P^1 elements (bottom).

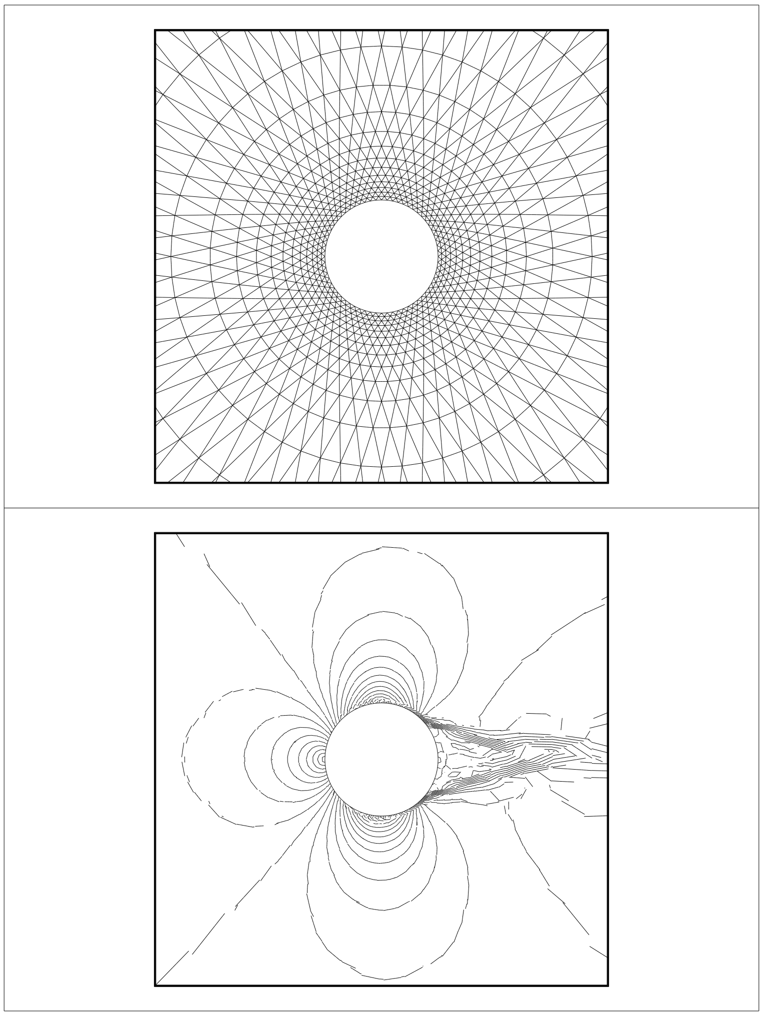


FIGURE 16. Grid "64×16" with a piecewise linear approximation of the circle (top) and the corresponding solution (Mach isolines) using P^1 elements (bottom).

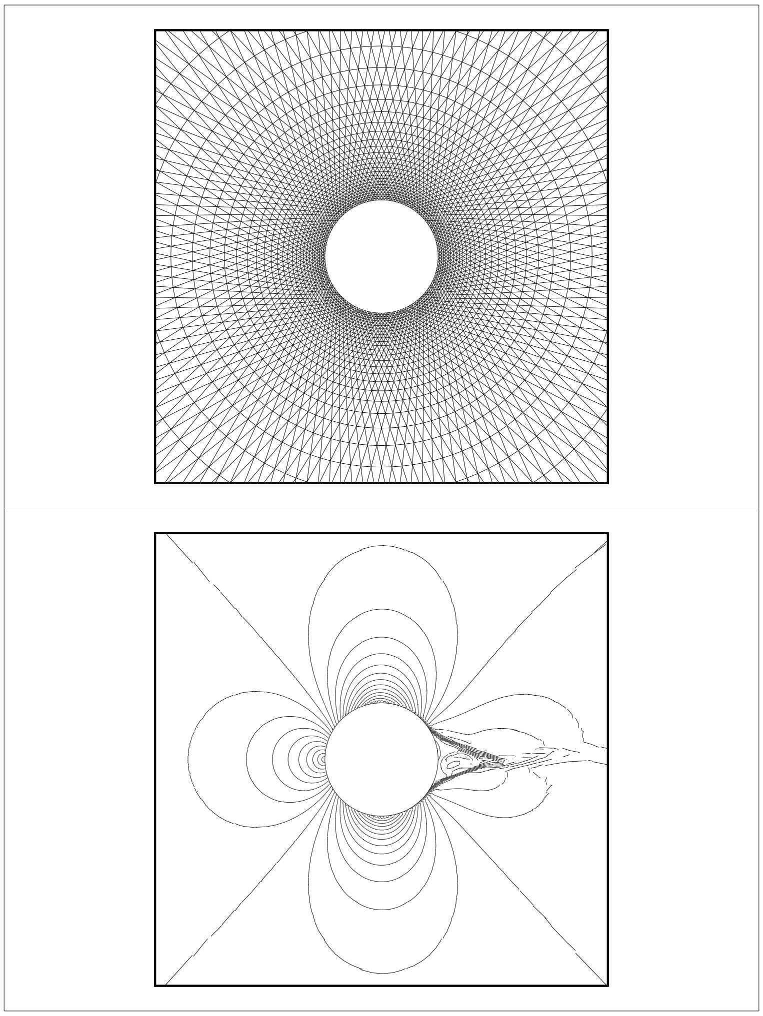


FIGURE 17. Grid “128×32” a piecewise linear approximation of the circle (top) and the corresponding solution (Mach isolines) using P^1 elements (bottom).

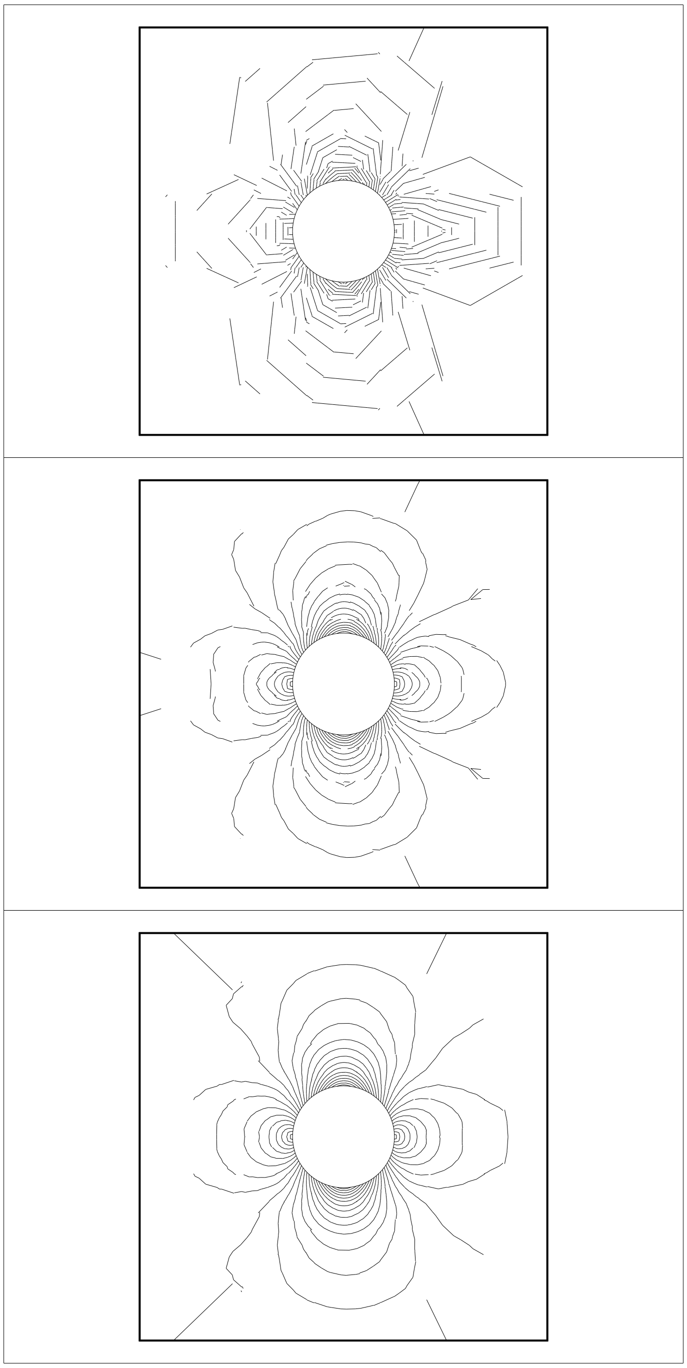


FIGURE 18. Grid “16×4” with exact rendering of the circle and the corresponding P^1 (top), P^2(middle), and P^3 (bottom) approximations (Mach isolines).

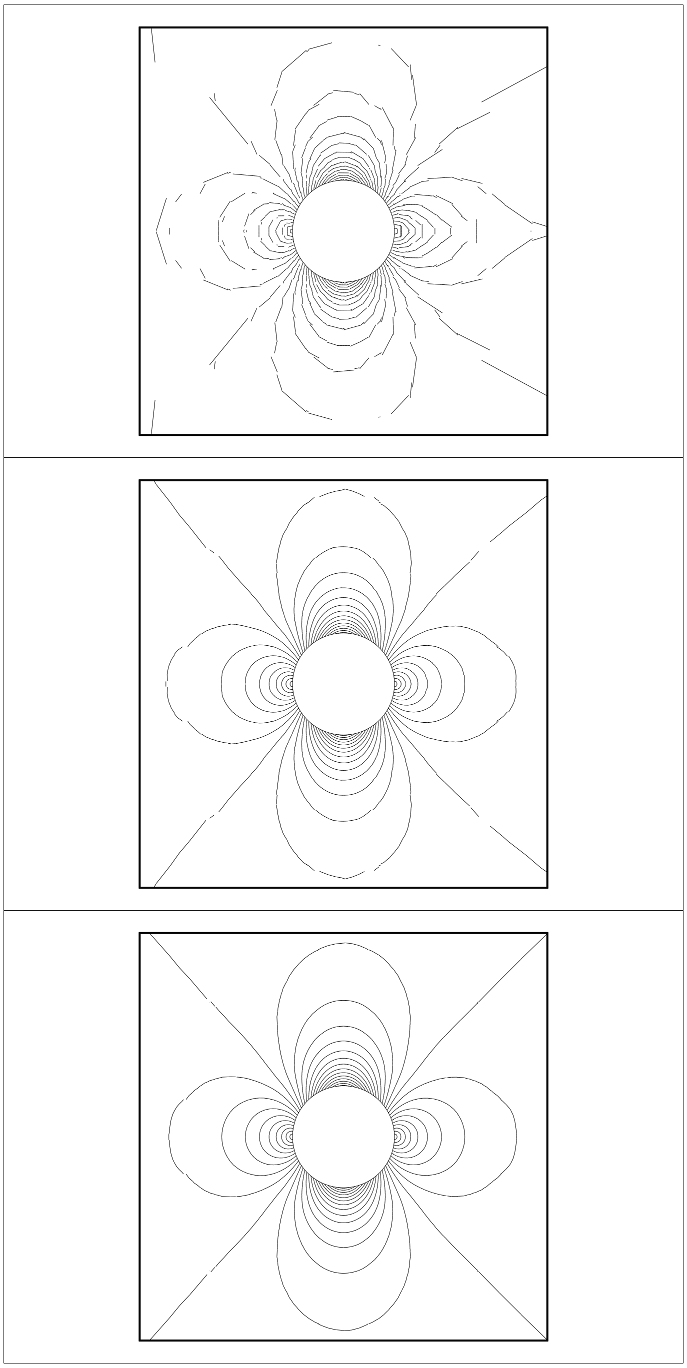


FIGURE 19. Grid "32 × 8" with exact rendering of the circle and the corresponding P^1 (top), P^2(middle), and P^3 (bottom) approximations (Mach isolines).

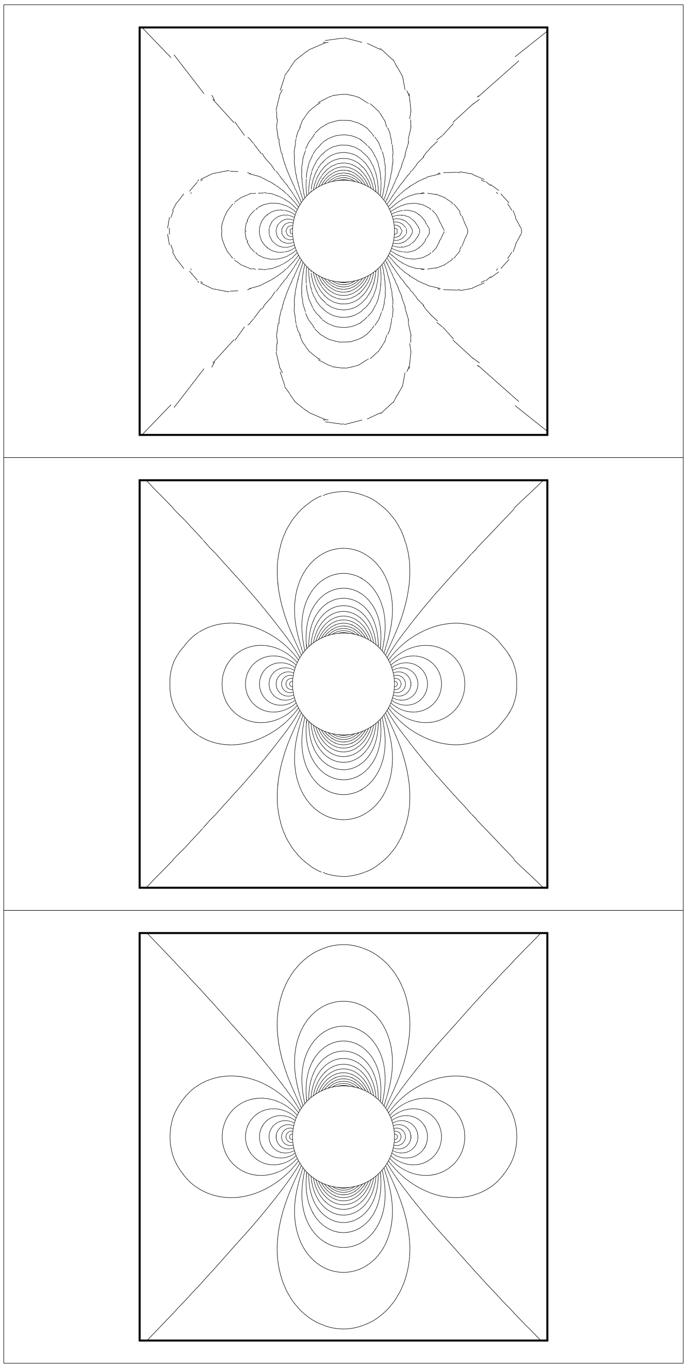


FIGURE 20. Grid "64×16" with exact rendering of the circle and the corresponding P^1 (top), P^2(middle), and P^3 (bottom) approximations (Mach isolines).

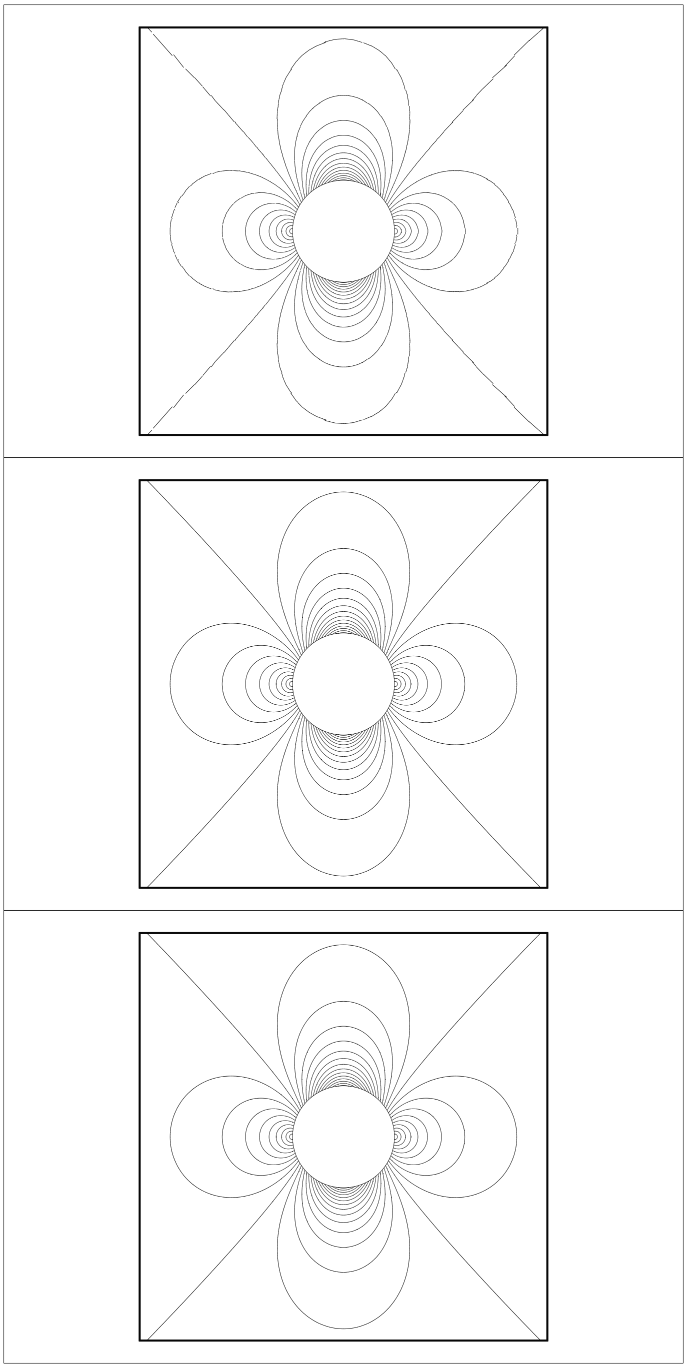


FIGURE 21. Grid "128×32" with exact rendering of the circle and the corresponding P^1 (top), P^2(middle), and P^3 (bottom) approximations (Mach isolines).

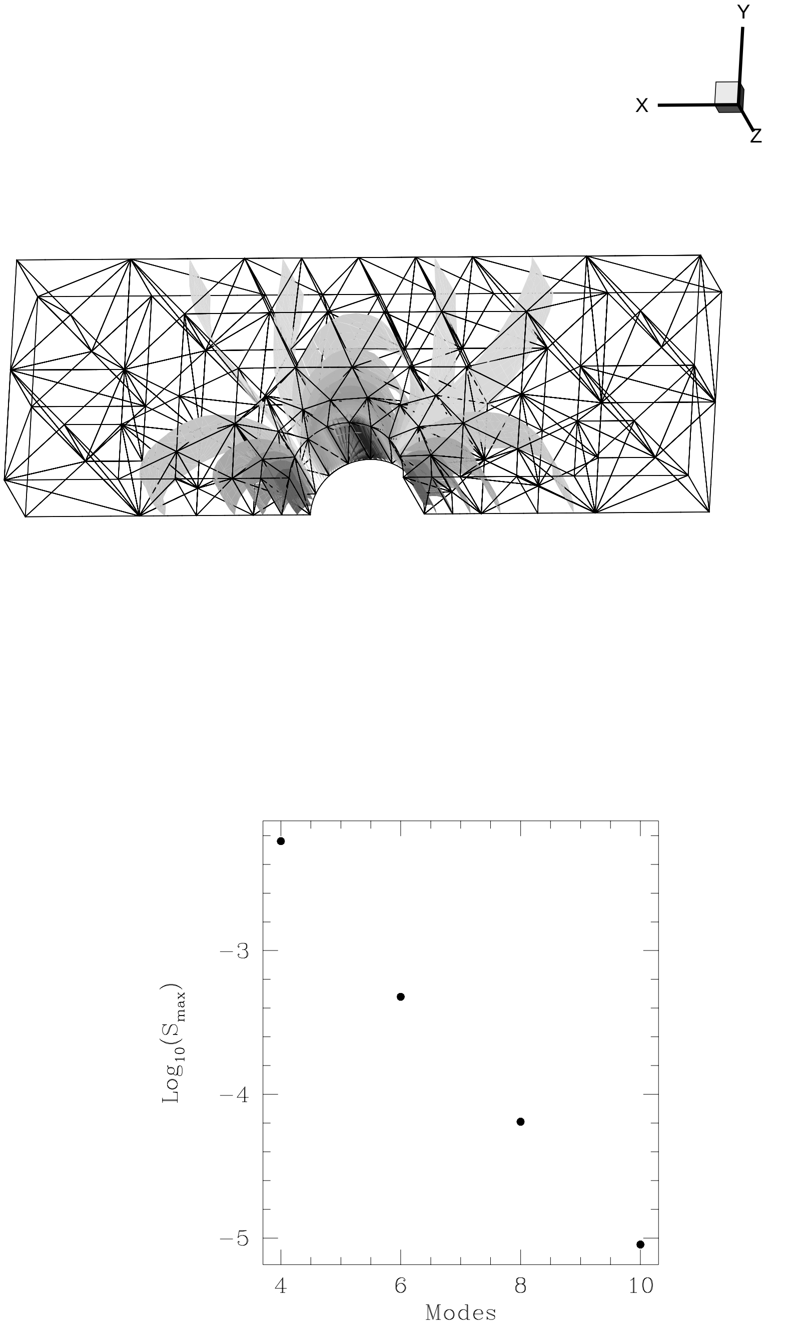


FIGURE 22. Three-dimensional flow over a semicircular bump. Mesh and density isosurfaces (top) and history of convergence with p-refinement of the maximum entropy generated (bottom). The degree of the polynomial plus one is plotted on the 'modes' axis.

CHAPTER 4

Convection-diffusion problems: The LDG method

4.1. Introduction

In this chapter, which follows the work by Cockburn and Shu [**18**], we restrict ourselves to the semidiscrete LDG methods for convection-diffusion problems with periodic boundary conditions. Our aim is to clearly display the most distinctive features of the LDG methods in a setting as simple as possible; the extension of the method to the fully discrete case is straightforward. In §2, we introduce the LDG methods for the simple one-dimensional case $d = 1$ in which

$$\mathbf{F}(u, Du) = f(u) - a(u)\,\partial_x u,$$

u is a scalar and $a(u) \geq 0$ and show some preliminary numerical results displaying the performance of the method. In this simple setting, the main ideas of how to device the method and how to analyze it can be clearly displayed in a simple way. Thus, the L^2-stability of the method is proven in the general nonlinear case and the rate of convergence of $(\Delta x)^k$ in the $\mathrm{L}^\infty(0,T;\mathrm{L}^2)$-norm for polynomials of degree $k \geq 0$ in the linear case is obtained; this estimate is sharp. In §3, we extend these results to the case in which u is a scalar and

$$\mathbf{F}_i(u, Du) = f_i(u) - \sum_{1\leq j\leq d} a_{ij}(u)\,\partial_{x_j} u,$$

where a_{ij} defines a positive semidefinite matrix. Again, the L^2-stability of the method is proven for the general nonlinear case and the rate of convergence of $(\Delta x)^k$ in the $\mathrm{L}^\infty(0,T;\mathrm{L}^2)$-norm for polynomials of degree $k \geq 0$ and arbitrary triangulations is proven in the linear case. In this case, the multidimensionality of the problem and the arbitrariness of the grids increase the technicality of the analysis of the method which, nevertheless, uses the same ideas of the one-dimensional case. In §4, the extension of the LDG method to multidimensional systems is briefly described some numerical results for the compressible Navier-Stokes equations from the paper by Bassi and Rebay [**3**] and from the paper by Lomtev and Karniadakis [**46**] are presented.

4.2. The LDG methods for the one-dimensional case

In this section, we present and analyze the LDG methods for the following simple model problem:

$$\partial_t\, u + \partial_x\, (f(u) - a(u)\,\partial_x\, u) = 0 \quad \text{in } (0,T) \times (0,1), \tag{4.2.1}$$
$$u(t=0) = u_0, \quad \text{on } (0,1), \tag{4.2.2}$$

with periodic boundary conditions.

4.2.1. General formulation and main properties. To define the LDG method, we introduce the new variable $q = \sqrt{a(u)}\,\partial_x u$ and rewrite the problem (4.2.1), (4.2.2) as follows:

$$\partial_t u + \partial_x (f(u) - \sqrt{a(u)}\,q) = 0 \quad \text{in } (0,T)\times(0,1), \tag{4.2.3}$$
$$q - \partial_x g(u) = 0 \quad \text{in } (0,T)\times(0,1), \tag{4.2.4}$$
$$u(t=0) = u_0, \quad \text{on } (0,1), \tag{4.2.5}$$

where $g(u) = \int^u \sqrt{a(s)}\,ds$. The LDG method for (4.2.1), (4.2.2) is now obtained by simply discretizing the above system with the Discontinuous Galerkin method.

To do that, we follow [**15**] and [**14**]. We define the flux $\mathbf{h} = (\,h_u, h_q\,)^t$ as follows:

$$\mathbf{h}(u,q) = (\,f(u) - \sqrt{a(u)}\,q\,,\,-g(u)\,)^t. \tag{4.2.6}$$

For each partition of the interval $(0,1)$, $\{\,x_{j+1/2}\,\}_{j=0}^{N}$, we set $I_j = (x_{j-1/2}, x_{j+1/2})$, and $\Delta x_j = x_{j+1/2} - x_{j-1/2}$ for $j = 1,\ldots,N$; we denote the quantity $\max_{1\le j\le N}\,\Delta x_j$ by Δx . We seek an approximation $\mathbf{w}_h = (u_h, q_h)^t$ to $\mathbf{w} = (u,q)^t$ such that for each time $t \in [0,T]$, both $u_h(t)$ and $q_h(t)$ belong to the finite dimensional space

$$V_h = V_h^k = \{v \in L^1(0,1) : v|_{I_j} \in P^k(I_j),\ j = 1,\ldots,N\}, \tag{4.2.7}$$

where $P^k(I)$ denotes the space of polynomials in I of degree at most k. In order to determine the approximate solution (u_h, q_h), we first note that by multiplying (4.2.3), (4.2.4), and (4.2.5) by arbitrary, smooth functions v_u, v_q, and v_i, respectively, and integrating over I_j, we get, after a simple formal integration by parts in (4.2.3) and (4.2.4),

$$\begin{aligned}
&\int_{I_j} \partial_t u(x,t)\,v_u(x)\,dx - \int_{I_j} h_u(\mathbf{w}(x,t))\,\partial_x v_u(x)\,dx \\
&+h_u(\mathbf{w}(x_{j+1/2},t))\,v_u(x_{j+1/2}^-) - h_u(\mathbf{w}(x_{j-1/2},t))\,v_u(x_{j-1/2}^+) = 0, \qquad (4.2.8)\\
&\int_{I_j} q(x,t)\,v_q(x)\,dx - \int_{I_j} h_q(\mathbf{w}(x,t))\,\partial_x v_q(x)\,dx \\
&+h_q(\mathbf{w}(x_{j+1/2},t))\,v_q(x_{j+1/2}^-) - h_q(\mathbf{w}(x_{j-1/2},t))\,v_q(x_{j-1/2}^+) = 0, \qquad (4.2.9)\\
&\int_{I_j} u(x,0)\,v_i(x)\,dx = \int_{I_j} u_0(x)\,v_i(x)\,dx. \qquad (4.2.10)
\end{aligned}$$

Next, we replace the smooth functions v_u, v_q, and v_i by test functions $v_{h,u}$, $v_{h,q}$, and $v_{h,i}$, respectively, in the finite element space V_h and the exact solution $\mathbf{w} = (u,q)^t$ by the approximate solution $\mathbf{w}_h = (u_h, q_h)^t$. Since this function is discontinuous in each of its components, we must also replace the nonlinear flux $\mathbf{h}(\mathbf{w}(x_{j+1/2},t))$ by a numerical flux $\hat{\mathbf{h}}(\mathbf{w})_{j+1/2}(t) = (\hat{h}_u(\mathbf{w}_h)_{j+1/2}(t), \hat{h}_q(\mathbf{w}_h)_{j+1/2}(t))$ that will be suitably chosen later. Thus, the approximate solution given by the LDG method is defined as the solution of the following weak formulation:

$$\forall\, v_{h,u} \in P^k(I_j):$$
$$\int_{I_j} \partial_t\, u_h(x,t)\, v_{h,u}(x)\, dx - \int_{I_j} h_u(\mathbf{w}_h(x,t))\, \partial_x\, v_{h,u}(x)\, dx$$
$$+\hat{h}_u(\mathbf{w}_h)_{j+1/2}(t)\, v_{h,u}(x^-_{j+1/2}) - \hat{h}_u(\mathbf{w}_h)_{j-1/2}(t)\, v_{h,u}(x^+_{j-1/2}) = 0, \tag{4.2.11}$$
$$\forall\, v_{h,q} \in P^k(I_j):$$
$$\int_{I_j} q_h(x,t)\, v_{h,q}(x)\, dx - \int_{I_j} h_q(\mathbf{w}_h(x,t))\, \partial_x\, v_{h,q}(x)\, dx$$
$$+\hat{h}_q(\mathbf{w}_h)_{j+1/2}(t)\, v_{h,q}(x^-_{j+1/2}) - \hat{h}_q(\mathbf{w}_h)_{j-1/2}(t)\, v_{h,q}(x^+_{j-1/2}) = 0, \tag{4.2.12}$$
$$\forall\, v_{h,i} \in P^k(I_j):$$
$$\int_{I_j} u_h(x,0)\, v_{h,i}(x)\, dx = \int_{I_j} u_0(x)\, v_{h,i}(x)\, dx. \tag{4.2.13}$$

It only remains to choose the numerical flux $\hat{\mathbf{h}}(\mathbf{w}_h)_{j+1/2}(t)$. We use the notation:

$$[\,p\,] = p^+ - p^-, \quad \text{and} \quad \overline{p} = \frac{1}{2}(p^+ + p^-), \text{and } p^\pm_{j+1/2} = p(x^\pm_{j+1/2}).$$

To be consistent with the type of numerical fluxes used in the RKDG methods, we consider numerical fluxes of the form

$$\hat{\mathbf{h}}(\mathbf{w}_h)_{j+1/2}(t) \equiv \hat{\mathbf{h}}(\mathbf{w}_h(x^-_{j+1/2}, t), \mathbf{w}_h(x^+_{j+1/2}, t)),$$

that (i) are locally Lipschitz and consistent with the flux $\mathbf{h}$, (ii) allow for a local resolution of q_h in terms of u_h, (iii) reduce to an E-flux (see Osher [**51**]) when $a(\cdot) \equiv 0$, and that (iv) enforce the L^2-stability of the method.

To reflect the convection-diffusion nature of the problem under consideration, we write our numerical flux as the sum of a convective flux and a diffusive flux:

$$\hat{\mathbf{h}}(\mathbf{w}^-, \mathbf{w}^+) = \hat{\mathbf{h}}_{conv}(\mathbf{w}^-, \mathbf{w}^+) + \hat{\mathbf{h}}_{diff}(\mathbf{w}^-, \mathbf{w}^+). \tag{4.2.14}$$

The convective flux is given by

$$\hat{\mathbf{h}}_{conv}(\mathbf{w}^-, \mathbf{w}^+) = \big(\hat{f}(u^-, u^+), 0\big)^t, \tag{4.2.15}$$

where $\hat{f}(u^-, u^+)$ is any locally Lipschitz E-flux consistent with the nonlinearity f, and the diffusive flux is given by

$$\hat{\mathbf{h}}_{diff}(\mathbf{w}^-, \mathbf{w}^+) = \Big(-\frac{[\,g(u)\,]}{[\,u\,]}\,\overline{q},\ -\overline{g(u)}\,\Big)^t - \mathbb{C}_{diff}\,[\,\mathbf{w}\,], \tag{4.2.16}$$

where

$$\mathbb{C}_{diff} = \begin{pmatrix} 0 & c_{12} \\ -c_{12} & 0 \end{pmatrix}, \tag{4.2.17}$$

$$c_{12} = c_{12}(\mathbf{w}^-, \mathbf{w}^+) \quad \text{is locally Lipschitz,} \tag{4.2.18}$$

$$c_{12} \equiv 0 \quad \text{when } a(\cdot) \equiv 0. \tag{4.2.19}$$

We claim that this flux satisfies the properties (i) to (iv).

Let us prove our claim. That the flux $\hat{\mathbf{h}}$ is consistent with the flux $\mathbf{h}$ easily follows from their definitions. That $\hat{\mathbf{h}}$ is locally Lipschitz follows from the fact that $\hat{f}(\cdot,\cdot)$ is locally Lipschitz and from (4.2.17); we assume that $f(\cdot)$ and $a(\cdot)$ are locally Lipschitz functions, of course. Property (i) is hence satisfied.

That the approximate solution q_h can be resolved element by element in terms of u_h by using (4.2.12) follows from the fact that, by (4.2.16), the flux $\hat{h}_q = -\overline{g(u)} - c_{12}\,[\,u\,]$ is independent of q_h. Property (ii) is hence satisfied.

Property (iii) is also satisfied by (4.2.19) and by the construction of the convective flux.

To see that the property (iv) is satisfied, let us first rewrite the flux $\hat{\mathbf{h}}$ in the following way:

$$\hat{\mathbf{h}}(\mathbf{w}^-, \mathbf{w}^+) = \Big(\frac{[\,\phi(u)\,]}{[\,u\,]} - \frac{[\,g(u)\,]}{[\,u\,]}\,\overline{q},\ -\overline{g(u)}\,\Big)^t - \mathbb{C}[\,\mathbf{w}\,],$$

where

$$\mathbb{C} \;= \begin{pmatrix} c_{11} & c_{12} \\ -c_{12} & 0 \end{pmatrix}, \quad c_{11} = \frac{1}{[\,u\,]}\left(\frac{[\,\phi(u)\,]}{[\,u\,]} - \hat{f}(u^-, u^+)\right). \tag{4.2.20}$$

with $\phi(u)$ defined by $\phi(u) = \int^u f(s)\,ds$. Since $\hat{f}(\cdot,\cdot)$ is an E-flux,

$$c_{11} \;= \frac{1}{[\,u\,]^2}\int_{u^-}^{u^+} \big(f(s) - \hat{f}(u^-, u^+)\big)\,ds \geq 0,$$

and so, by (4.2.17), the matrix $\mathbb{C}$ is semipositive definite. The property (iv) follows from this fact and from the following result.

THEOREM 4.1. *We have,*

$$\tfrac{1}{2}\int_0^1 u_h^2(x,T)\,dx + \int_0^T\!\!\int_0^1 q_h^2(x,t)\,dx\,dt + \Theta_{T,\mathbb{C}}([\mathbf{w}_h]) \leq \tfrac{1}{2}\int_0^1 u_0^2(x)\,dx,$$

where

$$\Theta_{T,\mathbb{C}}([\mathbf{w}_h]) \;= \int_0^T \sum_{1\leq j\leq N} \Big\{ [\mathbf{w}_h(t)]^t \mathbb{C}\,[\mathbf{w}_h(t)] \Big\}_{j+1/2} dt.$$

For a proof, see [**18**]. Thus, this shows that the flux $\hat{\mathbf{h}}$ under consideration does satisfy the properties (i) to (iv)- as claimed.

Now, we turn to the question of the quality of the approximate solution defined by the LDG method. In the linear case $f' \equiv c$ and $a(\cdot) \equiv a$, from the above stability result and from the the approximation properties of the finite element space V_h, we can prove the following error estimate. We denote the $\mathrm{L}^2(0,1)$-norm of the ℓ-th derivative of u by $|\,u\,|_\ell$.

THEOREM 4.2. *Let* $\mathbf{e}$ *be the approximation error* $\mathbf{w} - \mathbf{w}_h$. *Then we have,*

$$\left\{ \int_0^1 |\, e_u(x,T)\,|^2\, dx + \int_0^T\!\!\int_0^1 |\, e_q(x,t)\,|^2\, dx\, dt + \Theta_{T,\mathbb{C}}([\mathbf{e}]) \right\}^{1/2} \le C\,(\Delta x)^k,$$

where $C = C(k, |\,u\,|_{k+1}, |\,u\,|_{k+2})$. *In the purely hyperbolic case* $a = 0$, *the constant* C *is of order* $(\Delta x)^{1/2}$. *In the purely parabolic case* $c = 0$, *the constant* C *is of order* Δx *for even values of* k *for uniform grids and for* $\mathbb{C}$ *identically zero.*

For a proof, see [**18**]. The above error estimate gives a suboptimal order of convergence, but it is sharp for the LDG methods. Indeed, Bassi *et al* [**4**] 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 $\mathbb{C}$ identically zero. The numerical results for a purely parabolic problem that will be displayed later lead to the same conclusions; see Table 5 in the section §2.b.

The error estimate is also sharp in that the optimal order of convergence of $k+1/2$ is recovered in the purely hyperbolic case, as expected. This improvement of the order of convergence is a reflection of the *semipositive definiteness* of the matrix $\mathbb{C}$, which enhances the stability properties of the LDG method. Indeed, since in the purely hyperbolic case

$$\Theta_{T,\mathbb{C}}([\mathbf{w}_h]) \quad = \int_0^T \sum_{1\le j\le N} \left\{ [u_h(t)]^t\, c_{11}\, [u_h(t)] \right\}_{j+1/2} dt,$$

the method enforces a control of the jumps of the variable u_h, as shown in Proposition lemenergy. This additional control is reflected in the improvement of the order of accuracy from k in the general case to $k+1/2$ in the purely hyperbolic case.

However, this can only happen in the purely hyperbolic case for the LDG methods. Indeed, since $c_{11} = 0$ for $c = 0$, the control of the jumps of u_h is not enforced in the purely parabolic case. As indicated by the numerical experiments of Bassi *et al.* [**4**] and those of section §2.b below, this can result in the effective degradation of the order of convergence. To remedy this situation, the control of the jumps of u_h in the purely parabolic case can be easily enforced by letting c_{11} be strictly positive if $|\,c\,| + |\,a\,| > 0$. Unfortunately, this is not enough to guarantee an improvement of the accuracy: an additional control on the jumps of q_h is required! This can be easily achieved by allowing the matrix $\mathbb{C}$ to be *symmetric and positive definite* when $a > 0$. In this case, the order of convergence of $k+1/2$ can be easily obtained for the general convection-diffusion case. However, this would force the matrix entry c_{22} to be nonzero and the property (ii) of local resolvability of q_h in terms of u_h would not be satisfied anymore. As a consequence, the high parallelizability of the LDG would be lost.

The above result shows how strongly the order of convergence of the LDG methods depend on the choice of the matrix $\mathbb{C}$. In fact, the numerical results of section §2.b in uniform grids indicate that with yet another choice of the matrix $\mathbb{C}$, see (4.3.21), the LDG method converges with the optimal order of $k+1$ in the general case. The analysis of this phenomenon constitutes the subject of ongoing work.

4.3. Numerical results in the one-dimensional case

In this section we present some numerical results for the schemes discussed in this paper. We will only provide results for the following one dimensional, linear convection diffusion equation

$$\begin{aligned} &\partial_t u + c\,\partial_x u - a\,\partial_x^2 u = 0 \quad \text{in } (0,T)\times(0,2\pi), \\ &u(t=0,x) = \sin(x), \quad \text{on } (0,2\pi), \end{aligned}$$

where c and $a \geq 0$ are both constants; periodic boundary conditions are used. The exact solution is $u(t,x) = e^{-at}\sin(x-ct)$. We compute the solution up to $T=2$, and use the LDG method with $\mathbb{C}$ defined by

$$\mathbb{C} = \begin{pmatrix} \frac{|c|}{2} & -\frac{\sqrt{a}}{2} \\ \frac{\sqrt{a}}{2} & 0 \end{pmatrix}. \tag{4.3.21}$$

We notice that, for this choice of fluxes, the approximation to the convective term cu_x is the standard upwinding, and that the approximation to the diffusion term $a\,\partial_x^2 u$ is the standard three point central difference, for the P^0 case. On the other hand, if one uses a central flux corresponding to $c_{12} = -c_{21} = 0$, one gets a spread-out five point central difference approximation to the diffusion term $a\,\partial_x^2 u$.

The LDG methods based on P^k, with $k = 1,2,3,4$ are tested. Elements with equal size are used. Time discretization is by the third-order accurate TVD Runge-Kutta method [**58**], with a sufficiently small time step so that error in time is negligible comparing with spatial errors. We list the L_∞ errors and numerical orders of accuracy, for u_h, as well as for its derivatives suitably scaled $\Delta x^m \partial_x^m u_h$ for $1 \leq m \leq k$, at the center of of each element. This gives the complete description of the error for u_h over the whole domain, as u_h in each element is a polynomial of degree k. We also list the L_∞ errors and numerical orders of accuracy for q_h at the element center.

In all the convection-diffusion runs with $a > 0$, accuracy of at least $(k+1)$-th order is obtained, for both u_h and q_h, when P^k elements are used. See Tables 1 to 3. The P^4 case for the purely convection equation $a = 0$ seems to be not in the asymptotic regime yet with $N = 40$ elements (further refinement with $N = 80$ suffers from round-off effects due to our choice of non-orthogonal basis functions), Table 4. However, the absolute values of the errors are comparable with the convection dominated case in Table 3.

Table 1. The heat equation $a = 1$, $c = 0$. L_∞ errors and numerical order of accuracy, measured at the center of each element, for $\Delta x^m \partial_x^m u_h$ for $0 \le m \le k$, and for q_h.

k	variable	$N = 10$	$N = 20$		$N = 40$	
		error	error	order	error	order
1	u	4.55E-4	5.79E-5	2.97	7.27E-6	2.99
	$\Delta x\, \partial_x u$	9.01E-3	2.22E-3	2.02	5.56E-4	2.00
	q	4.17E-5	2.48E-6	4.07	1.53E-7	4.02
2	u	1.43E-4	1.76E-5	3.02	2.19E-6	3.01
	$\Delta x\, \partial_x u$	7.87E-4	1.03E-4	2.93	1.31E-5	2.98
	$(\Delta x)^2\, \partial_x^2 u$	1.64E-3	2.09E-4	2.98	2.62E-5	2.99
	q	1.42E-4	1.76E-5	3.01	2.19E-6	3.01
3	u	1.54E-5	9.66E-7	4.00	6.11E-8	3.98
	$\Delta x\, \partial_x u$	3.77E-5	2.36E-6	3.99	1.47E-7	4.00
	$(\Delta x)^2\, \partial_x^2 u$	1.90E-4	1.17E-5	4.02	7.34E-7	3.99
	$(\Delta x)^3\, \partial_x^3 u$	2.51E-4	1.56E-5	4.00	9.80E-7	4.00
	q	1.48E-5	9.66E-7	3.93	6.11E-8	3.98
4	u	2.02E-7	5.51E-9	5.20	1.63E-10	5.07
	$\Delta x\, \partial_x u$	1.65E-6	5.14E-8	5.00	1.61E-9	5.00
	$(\Delta x)^2\, \partial_x^2 u$	6.34E-6	2.04E-7	4.96	6.40E-9	4.99
	$(\Delta x)^3\, \partial_x^3 u$	2.92E-5	9.47E-7	4.95	2.99E-8	4.99
	$(\Delta x)^4\, \partial_x^4 u$	3.03E-5	9.55E-7	4.98	2.99E-8	5.00
	q	2.10E-7	5.51E-9	5.25	1.63E-10	5.07

Table 2. The convection diffusion equation $a = 1$, $c = 1$. L_∞ errors and numerical order of accuracy, measured at the center of each element, for $\Delta x^m \partial_x^m u_h$ for $0 \leq m \leq k$, and for q_h.

k	variable	$N = 10$ error	$N = 20$ error	$N = 20$ order	$N = 40$ error	$N = 40$ order
1	u	6.47E-4	1.25E-4	2.37	1.59E-5	2.97
	$\Delta x\, \partial_x u$	9.61E-3	2.24E-3	2.10	5.56E-4	2.01
	q	2.96E-3	1.20E-4	4.63	1.47E-5	3.02
2	u	1.42E-4	1.76E-5	3.02	2.18E-6	3.01
	$\Delta x\, \partial_x u$	7.93E-4	1.04E-4	2.93	1.31E-5	2.99
	$(\Delta x)^2\, \partial_x^2 u$	1.61E-3	2.09E-4	2.94	2.62E-5	3.00
	q	1.26E-4	1.63E-5	2.94	2.12E-6	2.95
3	u	1.53E-5	9.75E-7	3.98	6.12E-8	3.99
	$\Delta x\, \partial_x u$	3.84E-5	2.34E-6	4.04	1.47E-7	3.99
	$(\Delta x)^2\, \partial_x^2 u$	1.89E-4	1.18E-5	4.00	7.36E-7	4.00
	$(\Delta x)^3\, \partial_x^3 u$	2.52E-4	1.56E-5	4.01	9.81E-7	3.99
	q	1.57E-5	9.93E-7	3.98	6.17E-8	4.01
4	u	2.04E-7	5.50E-9	5.22	1.64E-10	5.07
	$\Delta x\, \partial_x u$	1.68E-6	5.19E-8	5.01	1.61E-9	5.01
	$(\Delta x)^2\, \partial_x^2 u$	6.36E-6	2.05E-7	4.96	6.42E-8	5.00
	$(\Delta x)^3\, \partial_x^3 u$	2.99E-5	9.57E-7	4.97	2.99E-8	5.00
	$(\Delta x)^4\, \partial_x^4 u$	2.94E-5	9.55E-7	4.95	3.00E-8	4.99
	q	1.96E-7	5.35E-9	5.19	1.61E-10	5.06

Table 3. The convection dominated convection diffusion equation $a = 0.01$, $c = 1$. L_∞ errors and numerical order of accuracy, measured at the center of each element, for $\Delta x^m \partial_x^m u_h$ for $0 \le m \le k$, and for q_h.

k	variable	$N = 10$	$N = 20$		$N = 40$	
		error	error	order	error	order
1	u	7.14E-3	9.30E-4	2.94	1.17E-4	2.98
	$\Delta x\, \partial_x u$	6.04E-2	1.58E-2	1.93	4.02E-3	1.98
	q	8.68E-4	1.09E-4	3.00	1.31E-5	3.05
2	u	9.59E-4	1.25E-4	2.94	1.58E-5	2.99
	$\Delta x\, \partial_x u$	5.88E-3	7.55E-4	2.96	9.47E-5	3.00
	$(\Delta x)^2\, \partial_x^2 u$	1.20E-2	1.50E-3	3.00	1.90E-4	2.98
	q	8.99E-5	1.11E-5	3.01	1.10E-6	3.34
3	u	1.11E-4	7.07E-6	3.97	4.43E-7	4.00
	$\Delta x\, \partial_x u$	2.52E-4	1.71E-5	3.88	1.07E-6	4.00
	$(\Delta x)^2\, \partial_x^2 u$	1.37E-3	8.54E-5	4.00	5.33E-6	4.00
	$(\Delta x)^3\, \partial_x^3 u$	1.75E-3	1.13E-4	3.95	7.11E-6	3.99
	q	1.18E-5	7.28E-7	4.02	4.75E-8	3.94
4	u	1.85E-6	4.02E-8	5.53	1.19E-9	5.08
	$\Delta x\, \partial_x u$	1.29E-5	3.76E-7	5.10	1.16E-8	5.01
	$(\Delta x)^2\, \partial_x^2 u$	5.19E-5	1.48E-6	5.13	4.65E-8	4.99
	$(\Delta x)^3\, \partial_x^3 u$	2.21E-4	6.93E-6	4.99	2.17E-7	5.00
	$(\Delta x)^4\, \partial_x^4 u$	2.25E-4	6.89E-6	5.03	2.17E-7	4.99
	q	3.58E-7	3.06E-9	6.87	5.05E-11	5.92

Table 4. The convection equation $a = 0$, $c = 1$. L_∞ errors and numerical order of accuracy, measured at the center of each element, for $\Delta x^m \partial_x^m u_h$ for $0 \le m \le k$.

k	variable	$N = 10$	$N = 20$		$N = 40$	
		error	error	order	error	order
1	u	7.24E-3	9.46E-4	2.94	1.20E-4	2.98
	$\Delta x\, \partial_x u$	6.09E-2	1.60E-2	1.92	4.09E-3	1.97
2	u	9.96E-4	1.28E-4	2.96	1.61E-5	2.99
	$\Delta x\, \partial_x u$	6.00E-3	7.71E-4	2.96	9.67E-5	3.00
	$(\Delta x)^2\, \partial_x^2 u$	1.23E-2	1.54E-3	3.00	1.94E-4	2.99
3	u	1.26E-4	7.50E-6	4.07	4.54E-7	4.05
	$\Delta x\, \partial_x u$	1.63E-4	2.00E-5	3.03	1.07E-6	4.21
	$(\Delta x)^2\, \partial_x^2 u$	1.52E-3	9.03E-5	4.07	5.45E-6	4.05
	$(\Delta x)^3\, \partial_x^3 u$	1.35E-3	1.24E-4	3.45	7.19E-6	4.10
4	u	3.55E-6	8.59E-8	5.37	3.28E-10	8.03
	$\Delta x\, \partial_x u$	1.89E-5	1.27E-7	7.22	1.54E-8	3.05
	$(\Delta x)^2\, \partial_x^2 u$	8.49E-5	2.28E-6	5.22	2.33E-8	6.61
	$(\Delta x)^3\, \partial_x^3 u$	2.36E-4	5.77E-6	5.36	2.34E-7	4.62
	$(\Delta x)^4\, \partial_x^4 u$	2.80E-4	8.93E-6	4.97	1.70E-7	5.72

Finally, to show that the order of accuracy could really degenerate to k for P^k, as was already observed in [**4**], we rerun the heat equation case $a = 1, c = 0$ with the central flux

$$\mathbb{C} = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}. \tag{4.3.22}$$

This time we can see that the global order of accuracy in L_∞ is only k when P^k is used with an odd value of k.

Table 5. The heat equation $a = 1$, $c = 0$. L_∞ errors and numerical order of accuracy, measured at the center of each element, for $\Delta x^m \partial_x^m u_h$ for $0 \le m \le k$, and for q_h, using the central flux.

k	variable	$N = 10$	$N = 20$		$N = 40$	
		error	error	order	error	order
1	u	3.59E-3	8.92E-4	2.01	2.25E-4	1.98
	$\Delta x\, \partial_x u$	2.10E-2	1.06E-2	0.98	5.31E-3	1.00
	q	2.39E-3	6.19E-4	1.95	1.56E-4	1.99
2	u	6.91E-5	4.12E-6	4.07	2.57E-7	4.00
	$\Delta x\, \partial_x u$	7.66E-4	1.03E-4	2.90	1.30E-5	2.98
	$(\Delta x)^2\, \partial_x^2 u$	2.98E-4	1.68E-5	4.15	1.03E-6	4.02
	q	6.52E-5	4.11E-6	3.99	2.57E-7	4.00
3	u	1.62E-5	1.01E-6	4.00	6.41E-8	3.98
	$\Delta x\, \partial_x u$	1.06E-4	1.32E-5	3.01	1.64E-6	3.00
	$(\Delta x)^2\, \partial_x^2 u$	1.99E-4	1.22E-5	4.03	7.70E-7	3.99
	$(\Delta x)^3\, \partial_x^3 u$	6.81E-4	8.68E-5	2.97	1.09E-5	2.99
	q	1.54E-5	1.01E-6	3.93	6.41E-8	3.98
4	u	8.25E-8	1.31E-9	5.97	2.11E-11	5.96
	$\Delta x\, \partial_x u$	1.62E-6	5.12E-8	4.98	1.60E-9	5.00
	$(\Delta x)^2\, \partial_x^2 u$	1.61E-6	2.41E-8	6.06	3.78E-10	6.00
	$(\Delta x)^3\, \partial_x^3 u$	2.90E-5	9.46E-7	4.94	2.99E-8	4.99
	$(\Delta x)^4\, \partial_x^4 u$	5.23E-6	7.59E-8	6.11	1.18E-9	6.01
	q	7.85E-8	1.31E-9	5.90	2.11E-11	5.96

4.4. The LDG methods for the multi-dimensional case

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

$$\partial_t u + \sum_{1\le i\le d} \partial_{x_i} \big(f_i(u) - \sum_{1\le j\le d} a_{ij}(u)\, \partial_{x_j} u\big) = 0 \quad \text{in } (0,T)\times(0,1)^d, \tag{4.4.23}$$

$$u(t=0) = u_0, \quad \text{on } (0,1)^d, \tag{4.4.24}$$

with periodic boundary conditions. Essentially, the one-dimensional case and the multidimensional 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 of entries $a_{ij}(u)$, which is assumed to be *symmetric, semipositive definite* and the introduction of the variables q_ℓ, and the second is the treatment of arbitrary meshes.

To define the LDG method, we first notice that, since the matrix $a_{ij}(u)$ is assumed to be symmetric and semipositive definite, there exists a symmetric matrix $b_{ij}(u)$ such that

$$a_{ij}(u) \quad = \textstyle\sum_{1\le \ell\le d} b_{i\ell}(u)\, b_{\ell j}(u). \tag{4.4.25}$$

Then we define the new scalar variables $q_\ell = \sum_{1\le j\le d} b_{\ell j}(u)\,\partial_{x_j} u$ and rewrite the problem (4.4.23), (4.4.24) as follows:

$$\partial_t u + \sum_{1\le i\le d} \partial_{x_i} \big(f_i(u) - \sum_{1\le \ell\le d} b_{i\ell}(u)\, q_\ell\big) = 0 \quad \text{in } (0,T)\times(0,1)^d, \tag{4.4.26}$$

$$q_\ell - \sum_{1\le j\le d} \partial_{x_j} g_{\ell j}(u) = 0, \quad \ell = 1,\dots d, \quad \text{in } (0,T)\times(0,1)^d, \tag{4.4.27}$$

$$u(t=0) = u_0, \quad \text{on } (0,1)^d, \tag{4.4.28}$$

where $g_{\ell j}(u) = \int^u b_{\ell j}(s)\,ds$. The LDG method is now obtained by discretizing the above equations by the Discontinuous Galerkin method.

We follow what was done in §2. So, we set $\mathbf{w} = (u,\mathbf{q})^t = (u, q_1,\cdots,q_d)^t$ and, for each $i = 1,\cdots,d$, introduce the flux

$$\mathbf{h}_i(\mathbf{w}) = \big(\, f_i(u) - \textstyle\sum_{1\le \ell\le d} b_{i\ell}(u)\, q_\ell, -g_{1i}(u),\cdots, -g_{di}(u)\,\big)^t. \tag{4.4.29}$$

We consider triangulations of $(0,1)^d$, $\mathbb{T}_{\Delta x} = \{\, K\,\}$, made of non-overlapping polyhedra. We require that for any two elements K and K', $\overline{K}\cap\overline{K}'$ is either a face e of both K and K' with nonzero $(d-1)$-Lebesgue measure $|\,e\,|$, or has Hausdorff dimension less than $d-1$. We denote by $\mathbb{E}_{\Delta x}$ the set of all faces e of the border of K for all $K\in\mathbb{T}_{\Delta x}$. The diameter of K is denoted by Δx_K and the maximum Δx_K, for $K\in\mathbb{T}_{\Delta x}$ is denoted by Δx. We require, for the sake of simplicity, that the triangulations $\mathbb{T}_{\Delta x}$ be regular, that is, there is a constant independent of Δx such that

$$\frac{\Delta x_K}{\rho_K} \le \sigma \quad \forall\, K \in \mathbb{T}_{\Delta x},$$

where ρ_K denotes the diameter of the maximum ball included in K.

We seek an approximation $\mathbf{w}_h = (u_h, \mathbf{q}_h)^t = (u_h, q_{h1}, \cdots, q_{hd})^t$ to $\mathbf{w}$ such that for each time $t \in [0,T]$, each of the components of $\mathbf{w}_h$ belong to the finite element space

$$V_h = V_h^k = \{\, v \in L^1((0,1)^d) : \, v|_K \in P^k(K)\ \forall\, K \in \mathbb{T}_{\Delta x}\}, \tag{4.4.30}$$

where $P^k(K)$ denotes the space of polynomials of total degree at most k. In order to determine the approximate solution $\mathbf{w}_h$, we proceed exactly as in the one-dimensional case. This time, however, the integrals are made on each element K of the triangulation $\mathbb{T}_{\Delta x}$. We obtain the following weak formulation on each element K of the triangulation $\mathbb{T}_{\Delta x}$:

$$\begin{aligned}
&\textstyle\int_K \partial_t\, u_h(x,t)\, v_{h,u}(x)\, dx - \sum_{1\le i\le d} \int_K\, h_{i\,u}(\mathbf{w}_h(x,t))\, \partial_{x_i}\, v_{h,u}(x)\, dx \\
&\quad + \textstyle\int_{\partial K} \hat{h}_u(\mathbf{w}_h, \mathbf{n}_{\partial K})(x,t)\, v_{h,u}(x)\, d\,\Gamma(x) = 0, \qquad \forall\, v_{h,u} \in P^k(K), \quad (4.4.31)
\end{aligned}$$

$$\begin{aligned}
&\qquad\qquad\qquad \text{For } \ell = 1, \cdots, d: \\
&\textstyle\int_K\, q_{h\ell}(x,t)\, v_{h,q_\ell}(x)\, dx - \sum_{1\le j\le d} \int_K\, h_{j\,q_\ell}(\mathbf{w}_h(x,t))\, \partial_{x_j}\, v_{h,q_\ell}(x)\, dx \\
&\quad + \textstyle\int_{\partial K} \hat{h}_{q_\ell}(\mathbf{w}_h, \mathbf{n}_{\partial K})(x,t)\, v_{h,q_\ell}(x)\, d\,\Gamma(x) = 0, \quad \forall\, v_{h,q_\ell} \in P^k(K), \quad (4.4.32) \\
&\textstyle\int_K\, u_h(x,0)\, v_{h,i}(x)\, dx = \int_K\, u_0(x)\, v_{h,i}(x)\, dx, \qquad \forall\, v_{h,i} \in P^k(K), \quad (4.4.33)
\end{aligned}$$

where $\mathbf{n}_{\partial K}$ denotes the outward unit normal to the element K at $x \in \partial K$. It remains to choose the numerical flux $(\hat{h}_u, \hat{h}_{q_1}, \cdots, \hat{h}_{q_d})^t \equiv \hat{\mathbf{h}} \equiv \hat{\mathbf{h}}(\mathbf{w}_h, \mathbf{n}_{\partial K})(x,t)$.

As in the one-dimensional case, we require that the fluxes $\hat{\mathbf{h}}$ be of the form

$$\hat{\mathbf{h}}(\mathbf{w}_h, \mathbf{n}_{\partial K})(x) \equiv \hat{\mathbf{h}}(\mathbf{w}_h(x^{int_K}, t), \mathbf{w}_h(x^{ext_K}, t); \mathbf{n}_{\partial K}),$$

where $\mathbf{w}_h(x^{int_K})$ is the limit at x taken from the interior of K and $\mathbf{w}_h(x^{ext_K})$ the limit at x from the exterior of K, and consider fluxes that (i) are locally Lipschitz, conservative, that is,

$$\hat{\mathbf{h}}(\mathbf{w}_h(x^{int_K}), \mathbf{w}_h(x^{ext_K}); \mathbf{n}_{\partial K}) + \hat{\mathbf{h}}(\mathbf{w}_h(x^{ext_K}), \mathbf{w}_h(x^{int_K}); -\mathbf{n}_{\partial K}) = 0,$$

and consistent with the flux

$$\sum_{1\le i\le d} \mathbf{h}_i \, n_{\partial K,i},$$

(ii) allow for a local resolution of each component of $\mathbf{q}_h$ in terms of u_h *only*, (iii) reduce to an E-flux when $a(\cdot) \equiv 0$, and that (iv) enforce the L^2-stability of the method.

Again, we write our numerical flux as the sum of a convective flux and a diffusive flux:

$$\hat{\mathbf{h}} = \hat{\mathbf{h}}_{conv} + \hat{\mathbf{h}}_{diff},$$

where the convective flux is given by

$$\hat{\mathbf{h}}_{conv}(\mathbf{w}^-, \mathbf{w}^+; \mathbf{n}) = \big(\hat{f}(u^-, u^+; \mathbf{n}), 0\big)^t,$$

where $\hat{f}(u^-, u^+; \mathbf{n})$ is any locally Lipschitz E-flux which is conservative and consistent with the nonlinearity

$$\sum_{1\le i\le d} f_i(u)\, n_i,$$

and the diffusive flux $\hat{\mathbf{h}}_{diff}(\mathbf{w}^-, \mathbf{w}^+; \mathbf{n})$ is given by

$$\Big(- \sum_{1\le i,\ell\le d} \frac{[\, g_{i\ell}(u) \,]}{[\, u \,]} \, \overline{q_\ell}\, n_i, \; - \sum_{1\le i\le d} \overline{g_{i1}(u)}\, n_i, \cdots, - \sum_{1\le i\le d} \overline{g_{id}(u)}\, n_i \Big)^t - \mathbb{C}_{diff}\, [\, \mathbf{w} \,],$$

where

$$\mathbb{C}_{diff} = \begin{pmatrix} 0 & c_{12} & c_{13} & \cdots & c_{1d} \\ -c_{12} & 0 & 0 & \cdots & 0 \\ -c_{13} & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ -c_{1d} & 0 & 0 & \cdots & 0 \end{pmatrix},$$

$$c_{1j} = c_{1j}(\mathbf{w}^-, \mathbf{w}^+) \quad \text{is locally Lipschitz for } j = 1, \cdots, d,$$
$$c_{1j} \equiv 0 \quad \text{when } a(\cdot) \equiv 0 \quad \text{for } j = 1, \cdots, d.$$

We claim that this flux satisfies the properties (i) to (iv).

To prove that properties (i) to (iii) are satisfied is now a simple exercise. To see that the property (iv) is satisfied, we first rewrite the flux $\hat{\mathbf{h}}$ in the following way:

$$\Big(- \sum_{1\le i,\ell\le d} \frac{[\, g_{i\ell}(u)\,]}{[\,u\,]}\, \overline{q_\ell}\, n_i,\ - \sum_{1\le i\le d} \overline{g_{i1}(u)}\, n_i, \cdots,\ - \sum_{1\le i\le d} \overline{g_{id}(u)}\, n_i\, \Big)^t - \mathbb{C}[\,\mathbf{w}\,],$$

where

$$\mathbb{C} = \begin{pmatrix} c_{11} & c_{12} & c_{13} & \cdots & c_{1d} \\ -c_{12} & 0 & 0 & \cdots & 0 \\ -c_{13} & 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ -c_{1d} & 0 & 0 & \cdots & 0 \end{pmatrix},$$

$$c_{11} = \frac{1}{[\,u\,]}\Big(\textstyle\sum_{1\le i\le d} \dfrac{[\,\phi_i(u)\,]}{[\,u\,]}\, n_i - \hat{f}(u^-, u^+; \mathbf{n}) \Big),$$

where $\phi_i(u) = \int^u f_i(s)\, ds$. Since $\hat{f}(\cdot,\cdot;\mathbf{n})$ is an E-flux,

$$c_{11} = \frac{1}{[\,u\,]^2} \int_{u^-}^{u^+} \big(\textstyle\sum_{1\le i\le d} f_i(s)\, n_i - \hat{f}(u^-, u^+; \mathbf{n}) \big)\, ds \ge 0,$$

and so the matrix $\mathbb{C}$ is semipositive definite. The property (iv) follows from this fact and from the following result.

THEOREM 4.3. *We have,*

$$\frac{1}{2}\int_{(0,1)^d} u_h^2(x,T)\, dx + \int_0^T\!\!\int_{(0,1)^d} |\, \mathbf{q}_h(x,t)\,|^2\, dx\, dt + \Theta_{T,\mathbb{C}}([\mathbf{w}_h]) \le \frac{1}{2}\int_{(0,1)^d} u_0^2(x)\, dx,$$

where

$$\Theta_{T,\mathbb{C}}([\mathbf{w}_h]) = \int_0^T \textstyle\sum_{e\in\mathbb{E}_{\Delta x}} \int_e [\mathbf{w}_h(x,t)]^t \mathbb{C}\, [\mathbf{w}_h(x,t)]\, d\Gamma(x)\, dt.$$

We can also prove the following error estimate. We denote the integral over $(0,1)^d$ of the sum of the squares of all the derivatives of order $(k+1)$ of u by $|\,u\,|_{k+1}^2$.

THEOREM 4.4. *Let* $\mathbf{e}$ *be the approximation error* $\mathbf{w} - \mathbf{w}_h$. *Then we have, for arbitrary, regular grids,*

$$\Big\{ \int_{(0,1)^d} |\, e_u(x,T)\,|^2\, dx + \int_0^T\!\!\int_{(0,1)^d} |\, \mathbf{e}_q(x,t)\,|^2\, dx\, dt + \Theta_{T,\mathbb{C}}([\mathbf{e}]) \Big\}^{1/2} \le C\,(\Delta x)^k,$$

where $C = C(k, |\,u\,|_{k+1}, |\,u\,|_{k+2})$. *In the purely hyperbolic case* $a_{ij} = 0$, *the constant* C *is of order* $(\Delta x)^{1/2}$. *In the purely parabolic case* $c = 0$, *the constant* C *is of order* Δx *for even values of* k *and of order* 1 *otherwise for Cartesian products of uniform grids and for* $\mathbb{C}$ *identically zero provided that the local spaces* Q^k *are used instead of the spaces* P^k, *where* Q^k *is the space of tensor products of one dimensional polynomials of degree* k.

4.5. Extension to multidimensional systems

In this chapter, we have considered the so-called LDG methods for convection-diffusion problems. For scalar problems in multidimensions, we have shown that they are L^2-stable and that in the linear case, they are of order k if polynomials of order k are used. We have also shown that this estimate is sharp and have displayed the strong dependence of the order of convergence of the LDG methods on the choice of the numerical fluxes.

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 multidimensional 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 multidimensional 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 $\mathbf{q} = (\partial_{x_1} u, ..., \partial_{x_d} u)$ as the auxilary variables. Although with this choice, the L^2-stability result will not be available theoretically, this would not cause any problem in practical implementations.

4.6. Some numerical results

Next, we present some numerical results from the papers by Bassi and Rebay [**3**] and Lomtev and Karniadakis [**46**].

• **Smooth, steady state solutions**. We start by displaying the convergence of the method for a p-refinement done by Lomtev and Karniadakis [**46**]. In Figure 1, we can see how the maximum errors in density, momentum, and energy decrease exponentially to zero as the degree k of the approximating polynomials increases while the grid is kept fixed; details about the exact solution can be found in [**46**].

Now, let us consider the laminar, transonic flow around the NACA0012 airfoil at an angle of attack of ten degrees, freestream Mach number $M = 0.8$, and Reynolds number (based on the freestream velocity and the airfoil chord) equal to 73; the wall temperature is set equal to the freestream total temperature. Bassy and Rebay [**3**] have computed the solution of this problem with polynomials of degree $1, 2$, and 3 and Lomtev and Karniadakis [**46**] have tried the same test problem with polynomials of degree $2, 4$, and 6 in a mesh of 592 elements which is about four times less elements than the mesh used by Bassi and Rebay [**3**]. In Figure 3, taken from [**46**], we display the pressure and drag coefficient distributions computed by Bassi and Rebay [**3**] with polynomials on degree 3 and the ones computed by Lomtev and Karniadakis [**46**] computed with polynomials of degree 6. We can see good agreement of both computations. In Figure 2, taken from [**46**], we see the mesh and the Mach isolines obtained with polynomials of degree two and four; note the improvement of the solution.

Next, we show a result from the paper by Bassi and Rebay [**3**]. We consider the laminar, subsonic flow around the NACA0012 airfoil at an angle of attack of zero degrees, freestream Mach number $M = 0.5$, and Reynolds number equal to

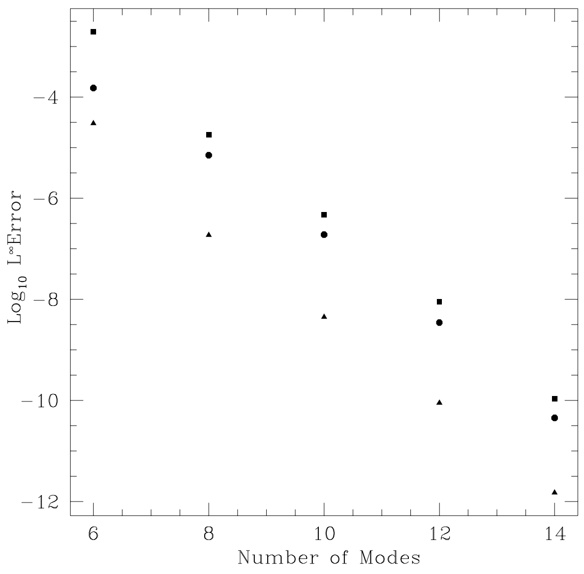


FIGURE 1. Maximum errors of the density (triangles), momemtum (circles) and energy (squares) as a function of the degree of the approximating polynomial plus one (called "number of modes" in the picture).

5000. In figure 4, we can see the Mach isolines corresponding to linear, quadratic, and cubic elements. In the figures 5, 6, and 7 details of the results with cubic elements are shown. Note how the boundary layer is captured withing a few layers of elements and how its separation at the trailing edge of the airfoil has been clearly resolved. Bassi and Rebay [**3**] report that these results are comparable to common structured and unstructures finite volume methods on much finer grids- a result consistent with the computational results we have displayed in these notes.

Finally, we present a not-yet-published result kindly provided by Lomtev and Karniadakis about the simulation of an expansion pipe flow. The smaller cylinder has a diameter of 1 and the larger cylinder has a diameter of 2. In Figure 8, we display the velocity profile and some streamlines for a Reynolds number equal to 50 and Mach number 0.2. The computation was made with polynomials of degree 5 and a mesh of 600 tetrahedra; of course the tetrahedra have curved faces to accomodate the exact boundaries. In Figure 9, we display a comparison between computational and experimental results. As a function of the Reynolds number, two quantities are plotted. The first is the distance between the step and the center of the vertex (lower brach) and the second is the distance from the step to the separation point (upper branch). The computational results are obtained by the method under consideration with polynomials of degree 5 for the compressible Navier Stokes equations, and by a standard Galerkin formulation in terms of velocity-pressure (NEKTAR), by Sherwin and Karniadakis [**56**], or in terms of velocity-vorticity (IVVA), by Trujillo [**61**], for the *incompressible* Navier Stokes equations; results produced by the code called PRISM are also included, see Newmann [**48**]. The experimental data was taken from Macagno and Tung [**49**]. The agreement between computations and experiments is remarkable.

• **Unsteady solutions**. To end this chapter, we present the computation of an unsteady solution by Lomtev and Karniadakis [**46**]. The test problem is the classical problem of a flow around a cylinder in two space dimensions. The Reynolds number is $10,000$ and the Mach number 0.2.

In Figure 10, the streamlines are shown for a computation made on a grid of 680 triangles (with curved sides fitting the cylinder) and polynomials whose degree could vary from element to element; the maximum degree was 5. In Figure 11, details of the mesh and the density around the cylinder are shown. Note how the method is able to capture the shear layer instability observed experimentally. For more details, see [**46**].

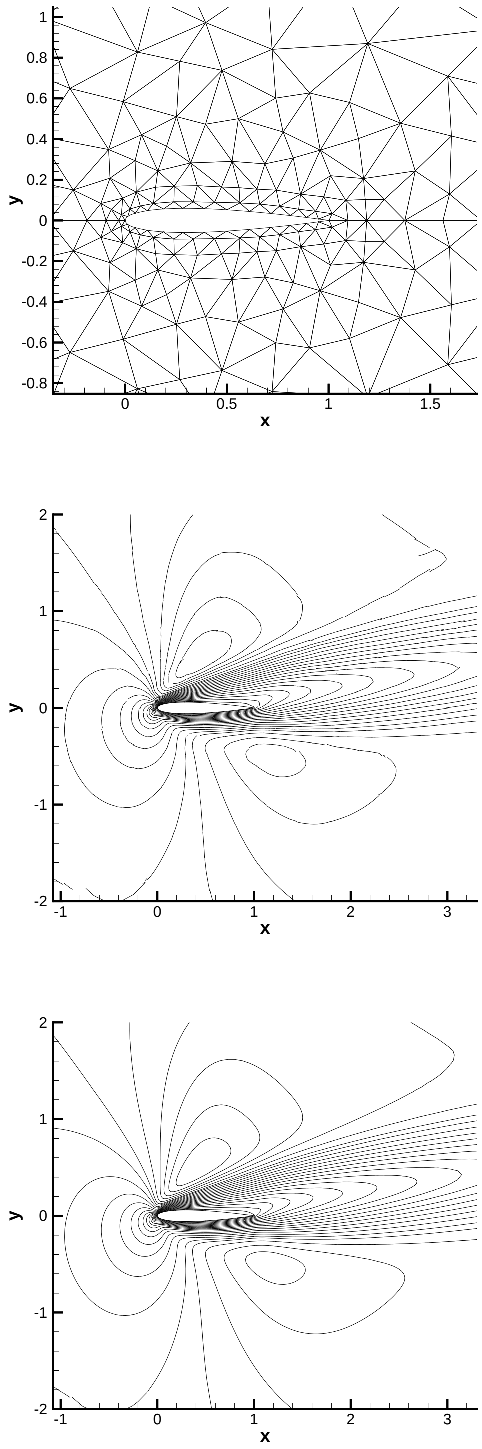


FIGURE 2. Mesh (top) and Mach isolines around the NACA0012 airfoil, ($Re = 73, M = 0.8$, angle of attack of ten degrees) for quadratic (middle) and quartic (bottom) elements.

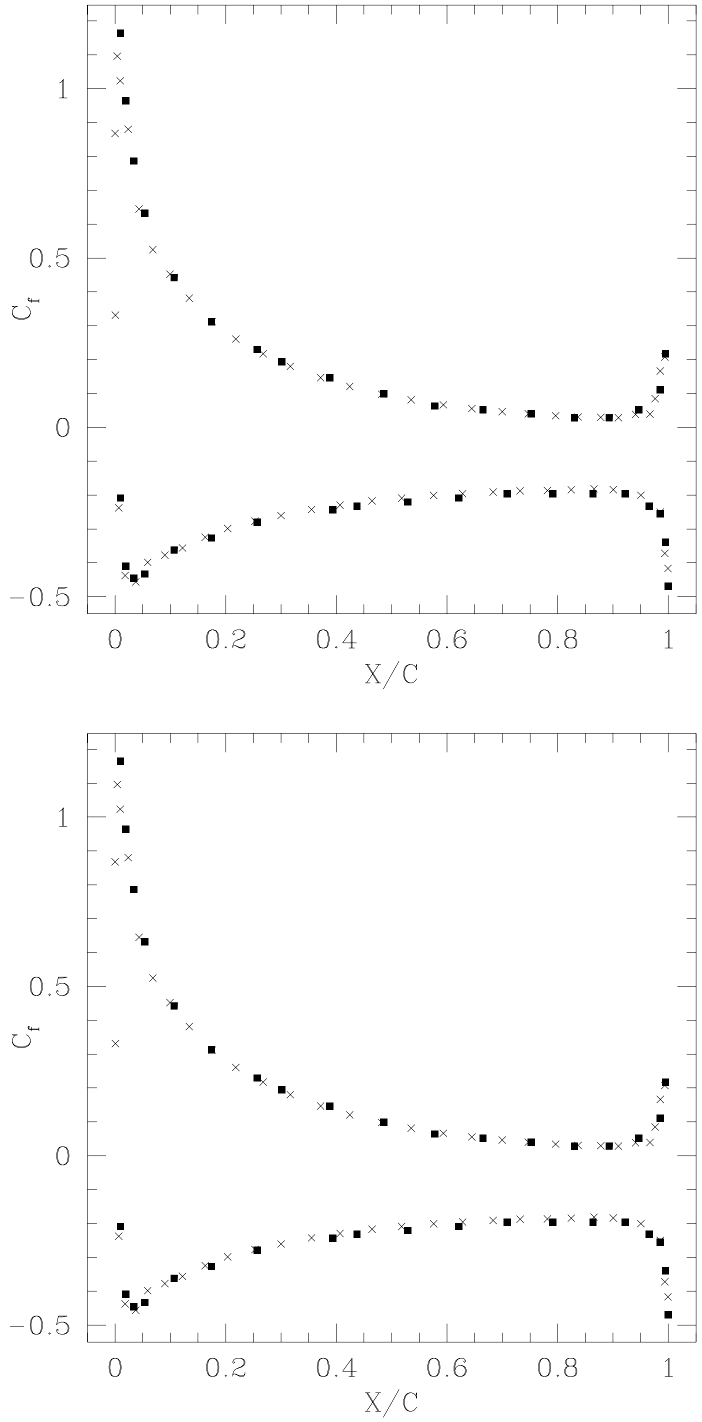


FIGURE 3. Pressure (top) and drag(bottom) coefficient distributions. The squares were obtained by Bassi and Rebay [**3**] with cubics and the crosses by Lomtev and Karniadakis [**46**] with polynomials of degree 6.

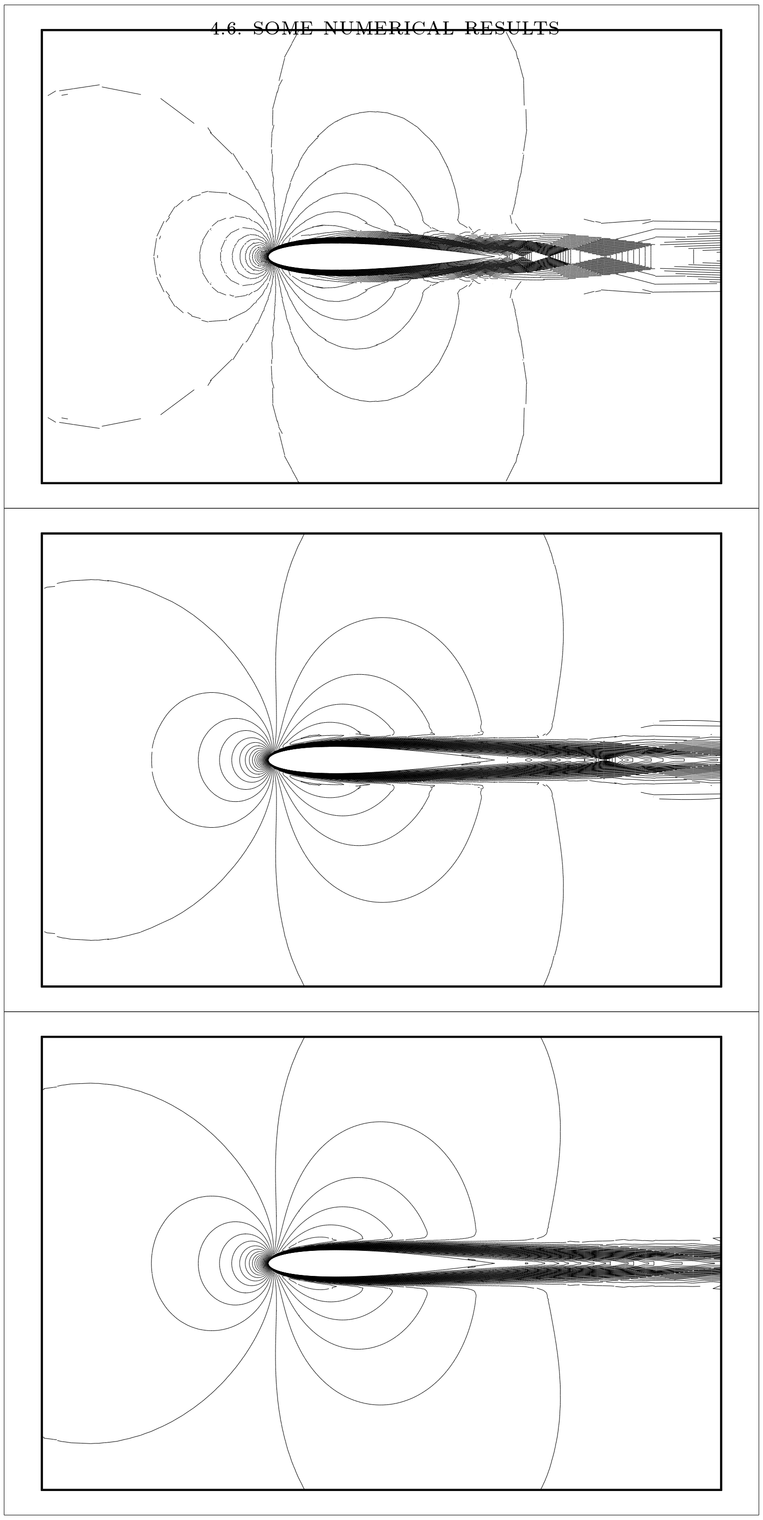


FIGURE 4. Mach isolines around the NACA0012 airfoil, ($Re = 5000, M = 0.5$, zero angle of attack) for the linear (top), quadratic (middle), and cubic (bottom) elements.

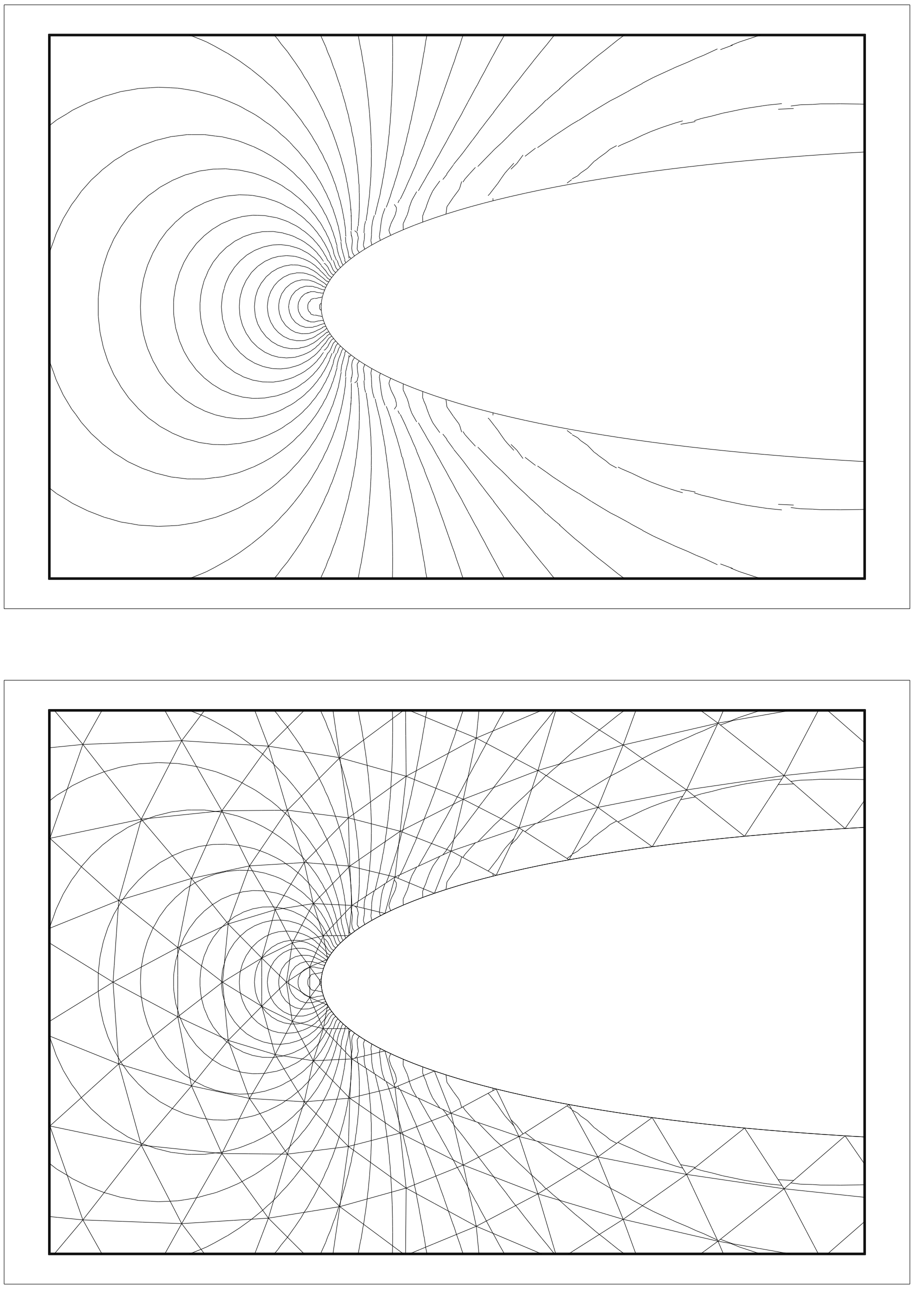


FIGURE 5. Pressure isolines around the NACA0012 airfoil, ($Re = 5000, M = 0.5$, zero angle of attack) for the for cubic elements without (top) and with (bottom) the corresponding grid.

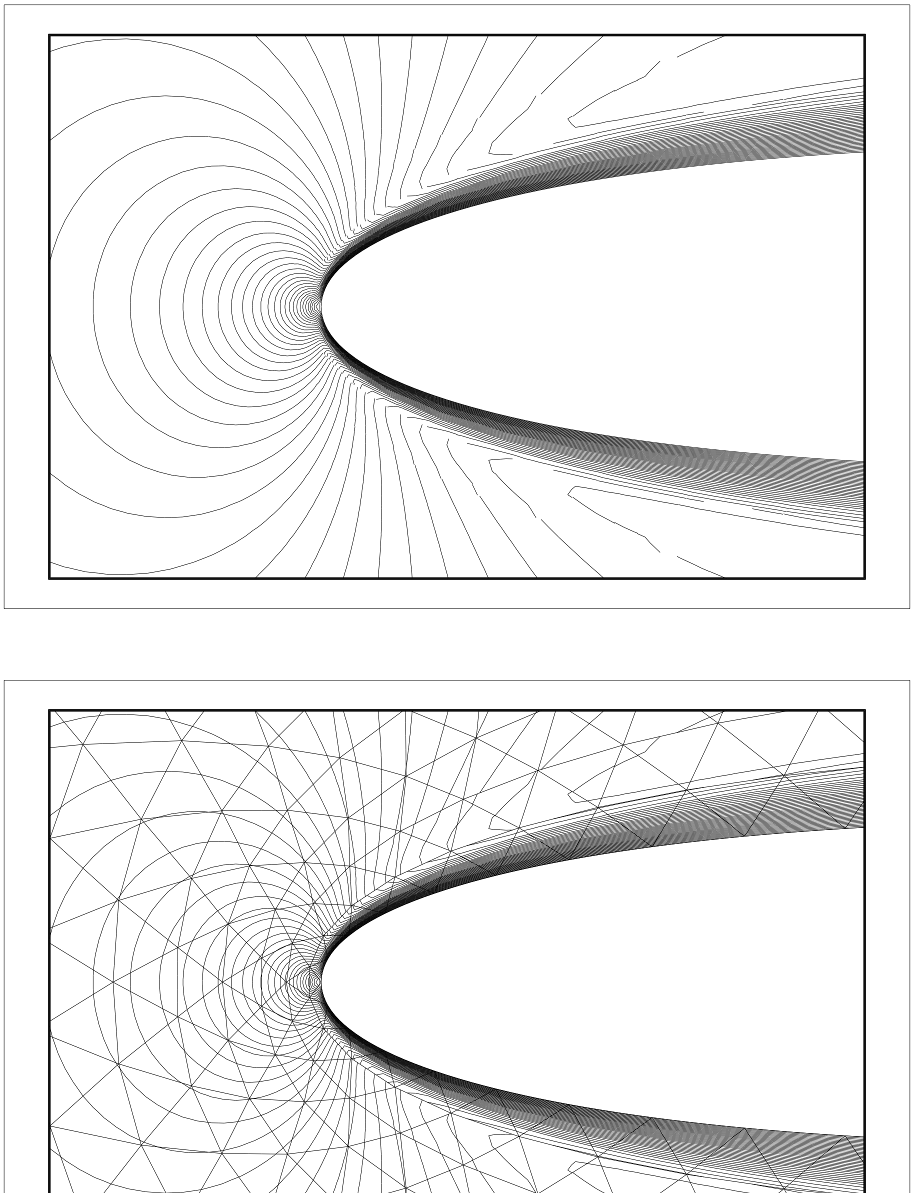
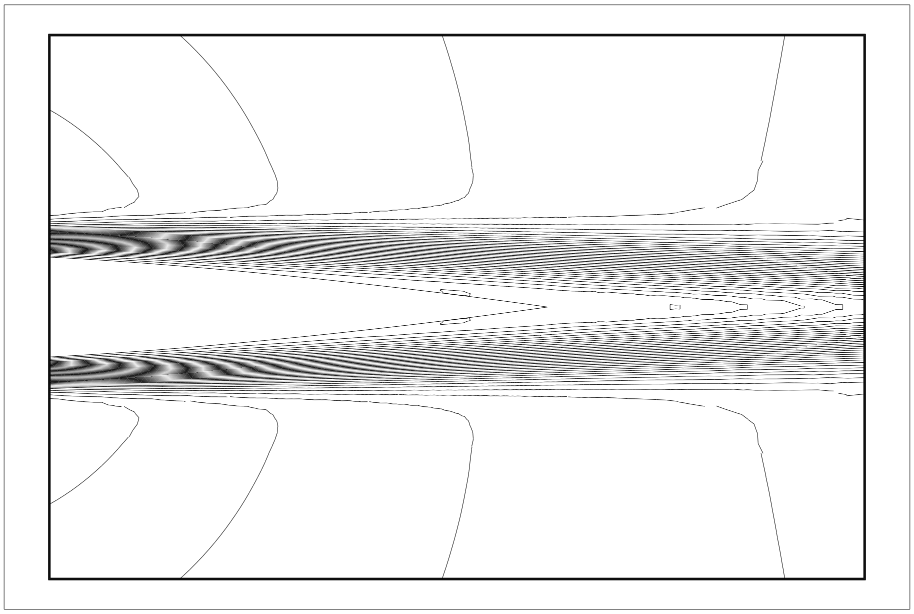


FIGURE 6. Mach isolines around the leading edge of the NACA0012 airfoil, ($Re = 5000, M = 0.5$, zero angle of attack) for the for cubic elements without (top) and with (bottom) the corresponding grid.



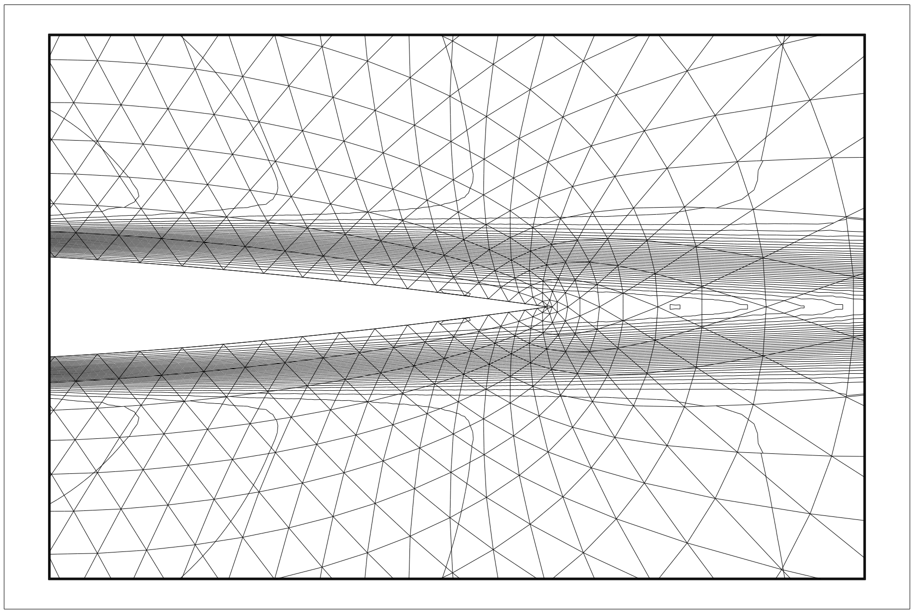


FIGURE 7. Mach isolines around the trailing edge of the NACA0012 airfoil, ($Re = 5000, M = 0.5$, zero angle of attack) for the for cubic elements without (top) and with (bottom) the corresponding grid.

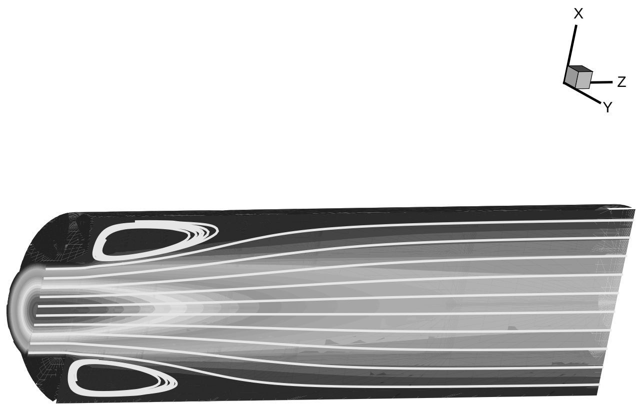


FIGURE 8. Expansion pipe flow at Reynolds number 50 and Mach number 0.2. Velocity profile and streamlines computed with a mesh of 600 elements and polynomials of degree 5.

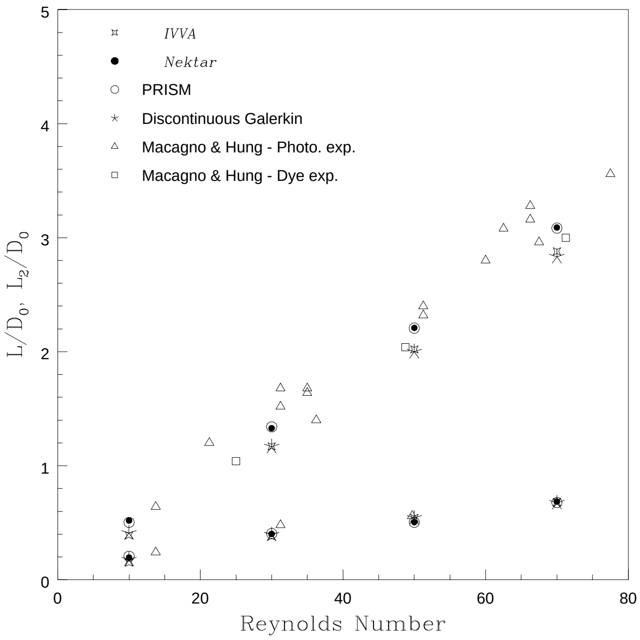


FIGURE 9. Expansion pipe flow: Comparison between computational and experimental results.

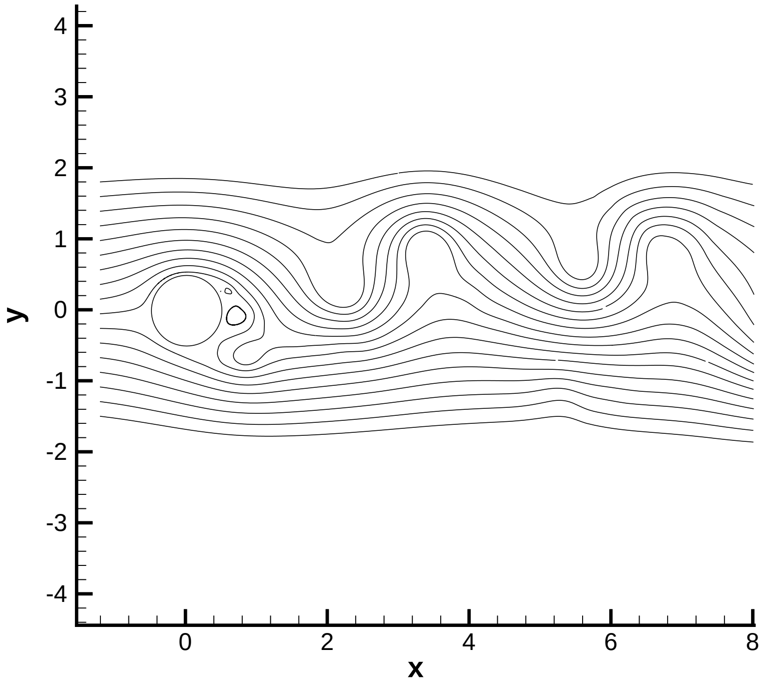
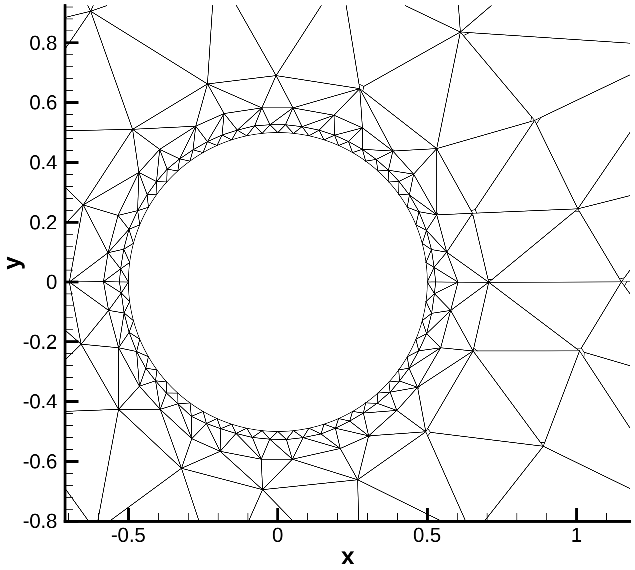


FIGURE 10. Flow around a cylinder with Reynolds number $10,000$ and Mach number 0.2. Streamlines. A mesh of 680 elements was used with polynomials that could change degree from element to element; the maximum degree was 5.



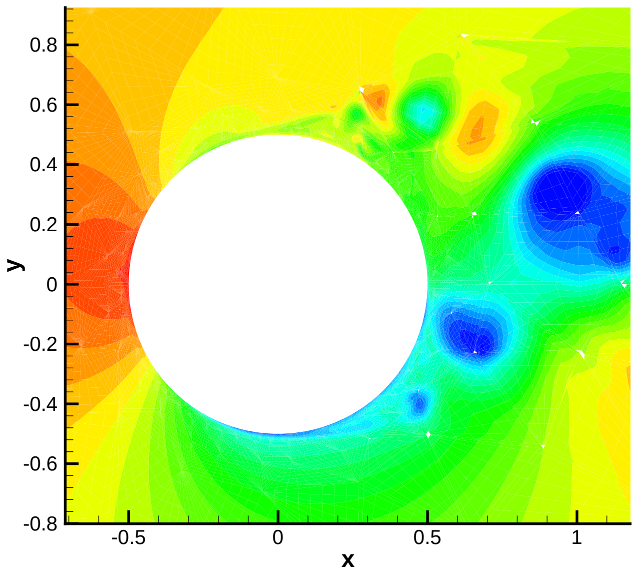


FIGURE 11. Flow around a cylinder with Reynolds number 10,000 and Mach number 0.2. Detail of the mesh (top) and density (bottom) around the cylinder.

Bibliography

[1] H.L. Atkins and C.-W. Shu. Quadrature-free implementation of discontinuous Galerkin methods for hyperbolic equations. *ICASE Report 96-51*, 1996. submitted to AIAA J.

[2] F. Bassi and S. Rebay. High-order accurate discontinuous finite element solution of the 2d Euler equations. *J. Comput. Phys.* to appear.

[3] F. Bassi and S. Rebay. A high-order accurate discontinuous finite element method for the numerical solution of the compressible Navier-Stokes equations. *J. Comput. Phys*, 131:267–279, 1997.

[4] F. Bassi, S. Rebay, M. Savini, G. Mariotti, and S. Pedinotti. A high-order accurate discontinuous finite element method for inviscid and viscous turbomachinery flows. *Proceedings of the Second European Conference ASME on Turbomachinery Fluid Dynamics and Thermodynamics*, 1995.

[5] K.S. Bey and J.T. Oden. A Runge-Kutta discontinuous Galerkin finite element method for high speed flows. *info AIAA* 10^{th} *Computational Fluid Dynamics Conference, Honolulu, Hawaii, June 24-27*, 1991.

[6] R. Biswas, K.D. Devine, and J. Flaherty. Parallel, adaptive finite element methods for conservation laws. *Applied Numerical Mathematics*, 14:255–283, 1994.

[7] G. Chavent and B. Cockburn. The local projection p^0 p^1-discontinuous-Galerkin finite element method for scalar conservation laws. M^2AN, 23:565–592, 1989.

[8] G. Chavent and G. Salzano. A finite element method for the 1d water flooding problem with gravity. *J. Comput. Phys*, 45:307–344, 1982.

[9] Z. Chen, B. Cockburn, C. Gardner, and J. Jerome. Quantum hydrodynamic simulation of hysteresis in the resonant tunneling diode. *J. Comput. Phys*, 117:274–280, 1995.

[10] Z. Chen, B. Cockburn, J. Jerome, and C.-W. Shu. Mixed-RKDG finite element method for the drift-diffusion semiconductor device equations. *VLSI Design*, 3:145–158, 1995.

[11] P. Ciarlet. *The finite element method for elliptic problems.* North Holland, 1975.

[12] B. Cockburn and P.-A. Gremaud. A priori error estimates for numerical methods for scalar conservation laws. part i: The general approach. *Math. Comp.*, 65:533–573, 1996.

[13] B. Cockburn, S. Hou, and C.W. Shu. Tvb Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws iv: The multidimensional case. *Math. Comp.*, 54:545–581, 1990.

[14] B. Cockburn, S.Y. Lin, and C.W. Shu. Tvb Runge-Kutta local projection discontinuous Galerkin finite element method for conservation laws iii: One dimensional systems. *J. Comput. Phys*, 84:90–113, 1989.

[15] B. Cockburn and C.W. Shu. Tvb Runge-Kutta local projection discontinuous Galerkin finite element method for scalar conservation laws ii: General framework. *Math. Comp.*, 52:411–435, 1989.

[16] B. Cockburn and C.W. Shu. The p^1-Rkdg method for two-dimensional Euler equations of gas dynamics. *ICASE Report No.91-32*, 1991.

[17] B. Cockburn and C.W. Shu. The Runge-Kutta local projection p^1-discontinuous Galerkin method for scalar conservation laws. M^2AN, 25:337–361, 1991.

[18] B. Cockburn and C.W. Shu. The local discontinuous Galerkin finite element method for convection-diffusion systems. *SIAM J. Numer. Anal.*. to appear.

[19] B. Cockburn and C.W. Shu. The Runge-Kutta discontinuous Galerkin finite element method for conservation laws v: Multidimensional systems. *J. Comput. Phys.*. to appear.

[20] H.L. deCougny, K.D. Devine, J.E. Flaherty, R.M. Loy, C. Ozturan, and M.S. Shephard. High-order accurate discontinuous finite element solution of the 2d Euler equations. *Applied Numerical Mathematics*, 16:157–182, 1994.

[21] K.D. Devine, J.E. Flaherty, R.M. Loy, and S.R. Wheat. Parallel partitioning strategies for the adaptive solution of conservation laws. *Rensselaer Polytechnic Institute Report No. 94-1*, 1994.

[22] K.D. Devine, J.E. Flaherty, S.R. Wheat, and A.B. Maccabe. A massively parallel adaptive finite element method with dynamic load balancing. *SAND 93-0936C*, 1993.

[23] K. Eriksson and C. Johnson. Adaptive finite element methods for parabolic problems i: A linear model problem. *SIAM J. Numer. Anal.*, 28:43–77, 1991.

[24] K. Eriksson and C. Johnson. Adaptive finite element methods for parabolic problems ii: Optimal error estimates in $l_\infty l_2$ and $l_\infty l_\infty$. *SIAM J. Numer. Anal.*, 32:706–740, 1995.

[25] K. Eriksson and C. Johnson. Adaptive finite element methods for parabolic problems iv: A nonlinear model problem. *SIAM J. Numer. Anal.*, 32:1729–1749, 1995.

[26] K. Eriksson and C. Johnson. Adaptive finite element methods for parabolic problems v: Long time integration. *SIAM J. Numer. Anal.*, 32:1750–1762, 1995.

[27] K. Eriksson, C. Johnson, and V. Thomée. Time discretization of parabolic problems by the discontinuous Galerkin method. *RAIRO, Anal. Numér.*, 19:611–643, 1985.

[28] J. Goodman and R. LeVeque. On the accuracy of stable schemes for 2d scalar conservation laws. *Math. Comp.*, 45:15–21, 1985.

[29] T. Hughes and A. Brook. Streamline upwind-Petrov-Galerkin formulations for convection dominated flows with particular emphasis on the incompressible navier-stokes equations. *Comp. Meth. in App. Mech. and Eng.*, 32:199–259, 1982.

[30] T. Hughes, L.P. Franca, M. Mallet, and A. Misukami. A new finite element formulation for computational fluid dynamics, i. *Comp. Meth. in App. Mech. and Eng.*, 54:223–234, 1986.

[31] T. Hughes, L.P. Franca, M. Mallet, and A. Misukami. A new finite element formulation for computational fluid dynamics, ii. *Comp. Meth. in App. Mech. and Eng.*, 54:341–355, 1986.

[32] T. Hughes, L.P. Franca, M. Mallet, and A. Misukami. A new finite element formulation for computational fluid dynamics, iii. *Comp. Meth. in App. Mech. and Eng.*, 58:305–328, 1986.

[33] T. Hughes, L.P. Franca, M. Mallet, and A. Misukami. A new finite element formulation for computational fluid dynamics, iv. *Comp. Meth. in App. Mech. and Eng.*, 58:329–336, 1986.

[34] T. Hughes and M. Mallet. A high-precision finite element method for shock-tube calculations. *Finite Element in Fluids*, 6:339–, 1985.

[35] P. Jamet. Galerkin-type approximations which are discontinuous in time for parabolic equations in a variable domain. *SIAM J. Numer. Anal.*, 15:912–928, 1978.

[36] G. Jiang and C.-W. Shu. On cell entropy inequality for discontinuous Galerkin methods. *Math. Comp.*, 62:531–538, 1994.

[37] C. Johnson and J. Pitkaranta. An analysis of the discontinuous Galerkin method for a scalar hyperbolic equation. *Math. Comp.*, 46:1–26, 1986.

[38] C. Johnson and J. Saranen. Streamline diffusion methods for problems in fluid mechanics. *Math. Comp.*, 47:1–18, 1986.

[39] C. Johnson and A. Szepessy. On the convergence of a finite element method for a non-linear hyperbolic conservation law. *Math. Comp.*, 49:427–444, 1987.

[40] C. Johnson, A. Szepessy, and P. Hansbo. On the convergence of shock capturing streamline diffusion finite element methods for hyperbolic conservation laws. *Math. Comp.*, 54:107–129, 1990.

[41] P. LeSaint and P.A. Raviart. On a finite element method for solving the neutron transport equation. *Mathematical aspects of finite elements in partial differential equations (C. de Boor, Ed.), Academic Press*, pages 89–145, 1974.

[42] W. B. Lindquist. Construction of solutions for two-dimensional riemann problems. *Comp. & Maths. with Appls.*, 12:615–630, 1986.

[43] W. B. Lindquist. The scalar Riemann problem in two spatial dimensions: piecewise smoothness of solutions and its breakdown. *SIAM J. Numer. Anal.*, 17:1178–1197, 1986.

[44] I. Lomtev and G.E. Karniadakis. A discontinuous spectral/hp element Galerkin method for the Navier-Stokes equations on unstructured grids. *Proc. IMACS WC'97*, Berlin, Germany, 1997.

[45] I. Lomtev and G.E. Karniadakis Simulations of viscous supersonic flows on unstructured h-p meshes. *AIAA 97-0754, 35th Aerospace Sciences Meeting*, Reno, 1997.

[46] I. Lomtev and G.E. Karniadakis A Discontinuous Galerkin Method for the Navier-Stokes equations. *Int. J. Num. Meth. Fluids*, submitted.

[47] I. Lomtev, C.B. Quillen and G.E. Karniadakis. Spectral/hp methods for viscous compressible flows on unstructured 2D meshes. *J. Comp. Phys.*, in press.

[48] D. Newmann. A Computational Study of Fluid/Structure Interactions: Flow-Induced Vibrations of a Flexible Cable Ph.D., Princeton, 1996.

[49] E.O. Macagno and T. Hung. Computational and experimental study of a captive annular eddy. *J.F.M.*, 28:43 –, 1967.

[50] X. Makridakis and I. Babusška. On the stability of the discontinuous Galerkin method for the heat equation. *SIAM J. Numer. Anal.*, 34:389–401, 1997.

[51] S. Osher. Riemann solvers, the entropy condition and difference approximations. *SIAM J. Numer. Anal.*, 21:217–235, 1984.

[52] C. Ozturan, H.L. deCougny, M.S. Shephard, and J.E. Flaherty. Parallel adaptive mesh refinement and redistribution on distributed memory computers. *Comput. Methods in Appl. Mech. and Engrg.*, 119:123–137, 1994.

[53] T. Peterson. A note on the convergence of the discontinuous Galerkin method for a scalar hyperbolic equation. *SIAM J. Numer. Anal.*, 28:133–140, 1991.

[54] W.H. Reed and T.R. Hill. Triangular mesh methods for the neutron transport equation. *Los Alamos Scientific Laboratory Report LA-UR-73-479*, 1973.

[55] G.R. Richter. An optimal-order error estimate for the discontinuous Galerkin method. *Math. Comp.*, 50:75–88, 1988.

[56] S.J. Sherwin and G. Karniadakis Tetrahedral hp finite elements: Algorithms and flow simulations. *J. Comput. Phys*, 124:314–45, 1996.

[57] C.-W. Shu and S. Osher. Efficient implementation of essentially non-oscillatory shock-capturing schemes. *J. Comput. Phys*, 77:439–471, 1988.

[58] C.-W. Shu and S. Osher. Efficient implementation of essentially non-oscillatory shock capturing schemes, ii. *J. Comput. Phys*, 83:32–78, 1989.

[59] C.W. Shu. TVB uniformly high order schemes for conservation laws. *Math. Comp.*, 49:105–121, 1987.

[60] C.W. Shu. TVD time discretizations. *SIAM J. Sci. Stat. Comput.*, 9:1073–1084, 1988.

[61] J.R. Trujillo. Effective High-Order Vorticity-Velocity Formulation. Ph.D., Princeton, 1997.

[62] B. van Leer. Towards the ultimate conservation difference scheme, ii. *J. Comput. Phys*, 14:361–376, 1974.

[63] B. van Leer. Towards the ultimate conservation difference scheme, v. *J. Comput. Phys*, 32:1–136, 1979.

[64] D. Wagner. The Riemann problem in two space dimensions for a single conservation law. *SIAM J. Math. Anal.*, 14:534–559, 1983.

[65] T.C. Warburton, I. Lomtev, R.M. Kirby and G.E. Karniadakis. A discontinuous Galerkin method for the Navier-Stokes equations on hybrid grids. *Center for Fluid Mechanics # 97-14*, Division of Applied Mathematics, Brown University, 1997.

[66] P. Woodward and P. Colella. The numerical simulation of two-dimensional fluid flow with strong shocks. *J. Comput. Phys*, 54:115–173, 1984.

[67] T. Zhang and G.Q.. Chen. Some fundamental concepts about systems of two spatial dimensional conservation laws. *Acta Math. Sci. (English Ed.)*, 6:463–474, 1986.

[68] T. Zhang and Y.X. Zheng. Two dimensional Riemann problems for a single conservation law. *Trans. Amer. Math. Soc.*, 312:589–619,1989.