Computers & Fluids 98 (2014) 186–195

Contents lists available at ScienceDirect

Computers & Fluids

journal homepage: www.elsevier.com/locate/compfluid

A note on the numerical dissipation from high-order discontinuous finite element schemes

Antony Jameson [a], Guido Lodato [b,*]

[a] Department of Aeronautics & Astronautics, Stanford University, Durand Building, 496 Lomita Mall, Stanford, CA 94305-4035, USA
[b] Department of Energy & Propulsion, INSA de Rouen – CORIA, Avenue de l'Université BP 12, 76801 St. Etienne du Rouvray, France

ARTICLE INFO

Article history:
Received 25 July 2013
Received in revised form 9 November 2013
Accepted 18 January 2014
Available online 4 February 2014

Keywords:
Energy stable flux reconstruction schemes
Spectral difference schemes
Artificial dissipation

ABSTRACT

This paper analyzes in detail the numerical dissipation term embedded in high-order discontinuous finite element type discretizations with particular emphasis on numerical schemes that can be formulated from the flux reconstruction methodology (for instance the spectral difference or the nodal discontinuous Galerkin schemes). By introducing the error estimate for the polynomial reconstruction of the solution, an analytical expression is given for the numerical dissipation term arising from using a Lax–Friedrichs type (Toro, 2009) numerical flux at the element interfaces. It is shown that, although some fundamental differences exist in the numerical dissipation term when odd or even numbers of solution points (respectively, even or odd polynomial orders) are used to represent the solution in the element, the overall expected accuracy of the scheme is fully recovered.

1. Introduction

During the last decade there has been widespread interest in the Discontinuous Galerkin (DG) method for conservation laws, for which the theoretical basis has been provided in Refs. [5,4,3,6–8]. In this formulation the solution is represented in terms of basis polynomials of degree p in each element, which are not required to match at the element interfaces. Then the inner product of the residual and each basis function is required to vanish in every element. Using integration by parts to evaluate the inner product, the flux at the element interfaces is calculated from the left and right states by an upwind formula corresponding to an approximate Riemann solution as in a Finite Volume (FV) scheme. Thus the DG approach combines the flavors of finite element and FV methods. The expected order of accuracy is $n = p + 1$ using polynomial basis functions of degree p, but for linear problems on uniform meshes super accuracy of order $2p + 1$ can be attained. For three-dimensional simulations on hexahedral elements tensor products may be used. Then to support polynomials of degree p one needs $p + 1$ collocation points in each direction leading to $(p + 1)^3$ Degrees of Freedom (DoF) for each dependent variable in a single element. Furthermore, the stiffness of the system of equations increases with p.

The rapid growth of computational complexity of DG methods with increasing order has spurred the search for more efficient variants or alternatives. One such approach is the Nodal DG (NDG) scheme [10] in which the solution is represented by Lagrange interpolation at a set of collocation points in each element, and the quadratures required are pre-integrated to produce local mass and stiffness matrices. Another alternative is the Spectral Difference (SD) method, first put forward by Kopriva and Kolias [17], under the name *staggered grid multi domain method* and subsequently extended to both quadrilateral and triangular elements [18,27]. Like the NDG method, the SD method represents the solution by Lagrange interpolation at $n = p + 1$ collocation points in each element. The flux is then represented by a polynomial of degree $p + 1$ at $n + 1$ interlocking flux points which include the element boundaries, where a Riemann flux is used as in the NDG scheme. While the SD method has proved robust and productive in a variety of applications, doubts have been raised about its stability on simplex elements, and it can be weakly unstable in one dimension depending on the choice of flux collocation points.

The Flux Reconstruction (FR) method by Huynh [11,12] further simplifies the treatment of the equations in differential form. Instead of calculating the flux at a separate set of flux collocation points, Huynh proposed simply to modify the flux $f(u_h)$ calculated from the solution at the interior nodal points by corrections from the left and right boundaries of each element based on the difference between the Riemann flux at the interface and the flux value calculated from the internal solution polynomial in the element. These corrections are propagated from each boundary by polynomials of degree $p + 1$ which vanish at the opposite boundary. Thus the corrected flux is represented by a polynomial of degree $p + 1$,

* Corresponding author. Tel.: +33 232956633.
E-mail address: guido.lodato@insa-rouen.fr (G. Lodato).

http://dx.doi.org/10.1016/j.compfluid.2014.01.016
0045-7930/

so that its derivative is a polynomial of degree p, consistent with the polynomial representing the solution. For the linear case Huynh was able to show that by appropriate choices of the correction polynomial he could recover both the standard NDG scheme and the SD scheme, as well as a variety of hitherto unexplored variations which might have some potential advantages. He also used Fourier analysis to verify the stability of some of these schemes for third-order accuracy. In nonlinear cases the FR schemes are no longer exactly equivalent to NDG or SD.

Utilizing the FR approach of Huynh [11,12], it was proved by Jameson [14] that (for 1D linear advection) a particular SD method is stable for all orders of accuracy in a broken norm of Sobolev type. Recently, this result has been extended by Vincent et al. [24], who identified a class of FR schemes which are guaranteed to be linearly stable for norms of the form $\int \left[u_h^2 + c\left(u_h^{(p)}\right)^2\right] dx$ where c can be varied over a wide range including values which recover the NDG and SD schemes. The extension of the stability proof to simplex elements and to advection–diffusion type equations has also been recently addressed [2,15,16,25,26,28].

In this study, the numerical dissipation term accompanying the energy stable FR discretization is carefully analyzed from the analytical point of view. Note however, that the relevant estimate of the numerical dissipation term is quite general and applies to a broad range of discontinuous FE schemes for which similar interface flux formulations are adopted and provided that the relevant discrete solution satisfies a particular form of the energy estimate (*cf.* Eq. (19)), as it will be explained in the following sections. The SD scheme and the whole set of discretizations within the energy stable FR methodology fall into this category. This further insight into the analytical form of the numerical dissipation term allows to better understand some fundamental differences that occur when different polynomial orders are used to represent the solution.

Adopting a 1D energy stable FR solver and focusing on the particular value of the parameter c which recovers the SD discretization, numerical tests are then included in Section 5 to measure the impact of the above mentioned differences. Although these tests are carried out using one specific scheme within the FR class, the results are expected to be sufficiently general to reflect the behavior of a broader range of discretizations belonging to the FR family. For simplicity, all the developments and tests will be carried out in the 1D case, and hence, the relevant results can be at least generalized to quadrilateral (2D) and hexahedral (3D) elements by virtue of the tensor product.

2. Mathematical formulation

For completeness, the main building block of the FR and SD schemes are summarized in the present section (see Refs. [11,22] for more details). This will be useful in order to later introduce the relevant energy estimate, and to point out that similar conclusions can be drawn regarding the dissipative nature of these two schemes. The energy estimate, in particular, will enable a clear identification of the total contribution to numerical dissipation, which will be thoroughly analyzed in Sections 3 and 4. It is worthwhile recalling that, as anticipated in the introduction, both schemes can be obtained from the more general energy stable FR methodology [24]. Furthermore, the above mentioned energy estimate holds valid for every discontinuous finite element scheme within the energy stable FR family.

2.1. Domain discretization

Consider the 1D scalar conservation law

$$\partial_t u + \partial_x f = 0, \tag{1}$$

within an arbitrary domain Ω, where x is a spatial coordinate, t is time, $u = u(x,t)$ is a conserved scalar quantity and $f = f(u)$ is the flux of u in the x direction. The domain is partitioned into N distinct elements $\Omega_i = \{x \mid x \in (x_i : x_{i+1})\}$ such that

$$\Omega = \bigcup_{i=1}^{N} \Omega_i, \quad \text{and} \quad 0 = \bigcap_{i=1}^{N} \Omega_i. \tag{2}$$

In order to achieve an efficient implementation, all elements in the physical domain are transformed to a standard element described by local coordinates $\xi \in [-1:1]$. The relevant transformation can be written via the mapping

$$\xi = \Gamma_i(x) = 2\frac{x - x_i}{x_{i+1} - x_i} - 1, \tag{3}$$

$$x = \Gamma_i^{-1}(\xi) = \frac{1-\xi}{2}x_i + \frac{1+\xi}{2}x_{i+1}. \tag{4}$$

The governing equations in the physical domain are then transferred into the computational domain, and they take the form

$$\partial_t \hat{u} + \partial_\xi \hat{f} = 0, \tag{5}$$

where

$$\hat{u} = \hat{u}(\xi,t) = u_i(\Gamma_i^{-1}(\xi),t), \quad \text{and} \quad \hat{f} = \hat{f}(\xi,t) = f_i(\Gamma_i^{-1}(\xi),t)/J_i, \tag{6}$$

with $J_i = (x_{i+1} - x_i)/2$.

2.2. The FR numerical scheme

In order to construct a degree $p = n-1$ polynomial inside each standard element, a set of n solution points is defined. Although the linear analysis of FR schemes implies that stability is independent of solution point location, in a recent work, Jameson [16] pointed out that, for the non-linear case, the choice of locating the solution points at the Gauss–Legendre quadrature points is optimal in reducing aliasing errors and providing good conditioning. Accordingly, the n solution points are obtained as the roots of the equation

$$P_n(\xi) = \frac{2n-1}{n}\xi P_{n-1}(\xi) - \frac{n-1}{n}P_{n-2}(\xi) = 0 \tag{7}$$

where $P_n(\xi)$ is the Legendre polynomial of order $n, P_1(\xi) = \xi$ and $P_0(\xi) = 1$. The approximating polynomials for the solution can be built using a Lagrange base defined as

$$h_i(\xi) = \prod_{\substack{s=1 \\ s \neq i}}^{n} \frac{\xi - \xi_s}{\xi_i - \xi_s}, \tag{8}$$

which can be used to obtain the (discontinuous) reconstructed solution and fluxes in the standard element as

$$\hat{u}^{\mathrm{d}}(\xi) = \sum_{i=1}^{n} \hat{u}_i h_i(\xi), \tag{9}$$

$$\hat{f}^{\mathrm{d}}(\xi) = \sum_{i=1}^{n} \hat{f}_i h_i(\xi), \tag{10}$$

where $\hat{u}_i$ and $\hat{f}_i$ are the solution and the flux evaluated at the ith solution point, respectively. The superscript 'd' is adopted to indicate that the above functions are, in general, discontinuous across the elements. Using Eq. (9), in particular, the approximate solution is evaluated at either end of the standard element. These values are used in conjunction with analogous information from adjoining elements to calculate common numerical interface fluxes from a two-point (upwind biased) flux formula.

In order to construct a continuous flux polynomial of order $p+1$, a correction flux $\hat{f}^{\mathrm{c}}$ of order $p+1$ is added to the approxi-

mate discontinuous flux. The correction flux is constructed such as to satisfy the following requirements: (a) the sum of the correction flux and the discontinuous flux equals the common numerical fluxes at the element interfaces; (b) the corrected flux follows (in some sense) the approximate discontinuous flux in the interior of each element. These requirements are satisfied by defining the correction flux as

$$\hat{f}^c(\xi) = [f_L^* - \hat{f}^d(-1)]g_L(\xi) + [f_R^* - \hat{f}^d(1)]g_R(\xi), \tag{11}$$

where f_L^* and f_R^* are the numerical interface fluxes at the left and right element interfaces, respectively; $g_L(\xi)$ and $g_R(\xi)$ are suitable polynomials of order $p+1$ such that

$$g_L(-1) = g_R(1) = 1, \quad g_L(1) = g_R(-1) = 0, \quad g_L(\xi) = g_R(-\xi). \tag{12}$$

The total (continuous of order $p+1$) flux polynomial function is finally computed from Eqs. (10)–(11) as

$$\begin{aligned}\hat{f}(\xi) &= \hat{f}^d(\xi) + \hat{f}^c(\xi) \\ &= \sum_{i=1}^{n} \hat{f}_i h_i(\xi) + [f_L^* - \hat{f}^d(-1)]g_L(\xi) + [f_R^* - \hat{f}^d(1)]g_R(\xi).\end{aligned} \tag{13}$$

2.3. The SD numerical scheme

Within each standard element, two sets of points are defined, namely the solution points and the flux points, as schematically illustrated in Fig. 1 for a one-dimensional element. In order to construct a degree $p = n-1$ polynomial, the solution at n points are required. In a recent work, Jameson [14] utilized a flux reconstruction [11] formulation to prove that the SD method is stable for all orders of accuracy, provided that the interior flux points are placed at the Gauss–Legendre quadrature points. In the non-linear case, as already mentioned, Gauss–Legendre solution points provide minimal aliasing errors [16]. Accordingly, the n solution points are obtained as the roots of Eq. (7), and the flux points are selected to be the Gauss–Legendre quadrature points of order $n-1$ plus the two end points 0 and 1.

The approximating polynomials for the solution (n points) and the fluxes ($n+1$ points) can be built using Lagrange bases defined as

$$h_i(\xi) = \prod_{\substack{s=1 \\ s\neq i}}^{n} \frac{\xi - \xi_s}{\xi_i - \xi_s}, \quad l_{i+1/2}(\xi) = \prod_{\substack{s=0 \\ s\neq i}}^{n} \frac{\xi - \xi_{s+1/2}}{\xi_{i+1/2} - \xi_{s+1/2}}. \tag{14}$$

Hence, the reconstructed solution and fluxes in the standard element are, respectively,

$$\hat{u}(\xi) = \sum_{i=1}^{n} \hat{u}_i h_i(\xi), \tag{15}$$

$$\hat{f}(\xi) = \sum_{i=0}^{n} \hat{f}_{i+1/2} l_{i+1/2}(\xi), \tag{16}$$

where $\hat{u}_i$ is the solution evaluated at the ith solution point, and $\hat{f}_{i+1/2}$, with $i = 1, \ldots, n-1$, are the fluxes evaluated at the interior flux points from the interpolated states defined by the solution polynomial. The fluxes $\hat{f}_{1/2}$ and $\hat{f}_{n+1/2}$ are set equal to the numerical interface fluxes f_L^* and f_R^* at the left and right element interfaces, respectively. Hence, the (continuous of order $p+1$) flux polynomial function can be also written as

$$\hat{f}(\xi) = \sum_{i=1}^{n-1} \hat{f}_{i+1/2} l_{i+1/2}(\xi) + f_L^* l_{1/2}(\xi) + f_R^* l_{n+1/2}(\xi). \tag{17}$$

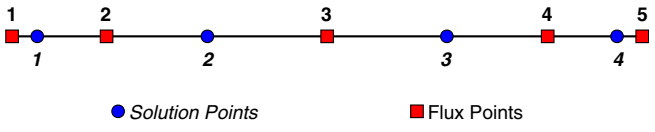


Fig. 1. Position of solution (circles) and flux (squares) points on the standard one-dimensional element for 3rd-order SD.

2.4. The energy estimate and the source of dissipation

For both the FR and SD schemes described above, the fluxes at the interior points are calculated (in some sense) from the solution polynomial, while the fluxes at each end-point is calculated as a common interface flux depending on the left and right values at the interface using a Lax–Friedrichs type [23] numerical flux. A Riemann solver is employed to compute the common inviscid flux at each cell interface to ensure both conservation and stability (*e.g.*, a Rusanov flux [21] or a Roe solver [20] with entropy fix [9]). The left and right states represent here the solution on both sides of the shared edge flux point. In Ref. [14] it was proved that NDG and SD schemes are linearly stable at any order of accuracy. In particular, let u_L and u_R be the values of $u(x)$ on the left and right sides of the cell interface, respectively. A Lax–Friedrichs type numerical flux for the linear case is defined as

$$f^* = \frac{1}{2}a(u_R + u_L) - \frac{1}{2}\alpha \mid a \mid (u_R - u_L), \quad 0 \leqslant \alpha \leqslant 1, \tag{18}$$

where the parameter α determines the amount of upwinding at the element interface. As demonstrated in Ref. [14], the discrete solution satisfies the energy estimate

$$\frac{\mathrm{d}}{\mathrm{d}t}\int_{x_1}^{x_{N+1}} \left[u^2 + \beta^{2p} c u^{(p)^2}\right] \mathrm{d}x \leqslant -\sum_{j=1}^{N-1} \left[\alpha \mid a \mid (u_R - u_L)^2\right]_{j+1/2} - \left[au^2\right]_{x_1}^{x_{N+1}}, \tag{19}$$

where the summation includes the contributions from all the interior interfaces, $\beta(x) = (x_{i+1} - x_i)/2 = J_i$ for $x \in (x_i : x_{i+1})$ is a piecewise constant scaling factor and c is a coefficient that depends on the polynomial order p and the relevant leading coefficient a_p:

$$c = \frac{2p}{2p+1} \frac{1}{p!(p+1)!} \frac{1}{a_p^2}. \tag{20}$$

Eq. (19) implies energy stability for the linear advection equation, and indicates that the total contribution to dissipation when Lax–Friedrichs type numerical fluxes are adopted comes from terms proportional to $(u_R - u_L)^2$, evaluated from left and right states at each element interface. Similar conclusions were later drawn for the whole family of FR schemes that can be obtained by a suitable choice of flux correction functions [24]. Linear stability for this particular family of FR schemes, which includes NDG and SD schemes as a special case, does not depend on the choice of the solution points. For the non-linear case, as mentioned, the use of Gauss–Legendre quadrature points is optimal in terms of aliasing errors and good conditioning [16]. In what follows, the error estimate for the polynomial reconstruction will be used to derive the analytical form of the dissipation term $\alpha \mid a \mid (u_R - u_L)/2$ in the case of equidistant solution points, and for the most optimal Gauss–Legendre quadrature points.

3. The numerical dissipation for equi-spaced solution points

In this section, the most simple and intuitive case of equidistant solution points within the standard element is first analyzed. Consider n equidistant solution points $\xi_1, \ldots, \xi_n$ and the corresponding (equidistant) flux points $\xi_{j+1/2}$ with $j = 0, \ldots, n$. From Eq. (A.7) the reconstructed solution can be expressed as

$$P_{n-1}(\xi) = u(\xi) - \frac{\omega_n(\xi)}{n!}u^{(n)}(\xi^*), \tag{21}$$

for some value ξ^* in the standard element. Note that the *hat* accent will be henceforth omitted under the implicit assumption that quantities are those associated with the standard transformed element in computational space. By introducing an *equivalent resolution* $h = 2/n$ (h represents the resolution a FV code would have if the standard element was filled with n identical cells), it is straightforward to show that

$$\omega_n(\xi)|_{\xi=1} = \frac{h}{2}\frac{3h}{2}\frac{5h}{2}\cdots\frac{(2n-1)h}{2} = K_u(n)h^n, \tag{22}$$

$$\omega_n(\xi)|_{\xi=-1} = \frac{-h}{2}\frac{-3h}{2}\frac{-5h}{2}\cdots\frac{-(2n-1)h}{2} = K_u(n)(-h)^n, \tag{23}$$

with

$$K_u(n) = \frac{1}{2^n}\prod_{j=1}^{n}(2j-1). \tag{24}$$

For n = 2, 3, 4 and 5, the extrapolated solution at the element interfaces are

$$P_1(-1) = u(-1) - \frac{3h^2}{8}u^{(2)}(\xi^*), \quad P_1(1) = u(1) - \frac{3h^2}{8}u^{(2)}(\xi^*), \tag{25}$$

$$P_2(-1) = u(-1) + \frac{5h^3}{16}u^{(3)}(\xi^*), \quad P_2(1) = u(1) - \frac{5h^3}{16}u^{(3)}(\xi^*), \tag{26}$$

$$P_3(-1) = u(-1) - \frac{35h^4}{128}u^{(4)}(\xi^*), \quad P_3(1) = u(1) - \frac{35h^4}{128}u^{(4)}(\xi^*), \tag{27}$$

$$P_4(-1) = u(-1) + \frac{63h^5}{256}u^{(5)}(\xi^*), \quad P_4(1) = u(1) - \frac{63h^5}{256}u^{(5)}(\xi^*). \tag{28}$$

Consider two contiguous elements; as mentioned, the numerical flux at the common interface is computed by a Lax–Friedrichs type formula, in which the left and right states are the rightmost extrapolated solution from the left element, and the leftmost extrapolated solution from the right element, respectively. Hence

$$u_L = [P_{n-1}(1)]_L, \quad \text{and} \quad u_R = [P_{n-1}(-1)]_R, \tag{29}$$

where the notation $[\cdot]_L$ (resp. $[\cdot]_R$) is a shorthand to indicate a quantity that is evaluated inside the left (resp. right) element. The numerical flux at the interface is evaluated as

$$f^* = \frac{1}{2}[f(u_R) + f(u_L)] - \frac{1}{2}\alpha \mid a \mid (u_R - u_L), \quad 0 \leqslant \alpha \leqslant 1, \tag{30}$$

where a is some suitable numerical approximation of the wave speed $\partial f/\partial u$. The second term on the RHS of Eq. (30) represents the dissipative flux d^* introduced by the scheme. Form Eqs. (21)–(29), this can be evaluated as

$$d^* = \frac{\alpha \mid a \mid}{2}\left\{[u(-1)]_R - [u(1)]_L + h^n\frac{K_u(n)}{n!}\left[u^{(n)}(\xi_L) - (-1)^n u^{(n)}(\xi_R)\right]\right\}, \tag{31}$$

where ξ_L and ξ_R are some locations inside the left and right elements, respectively. Note that, when the true solution is continuous across the elements, the difference $[u(-1)]_R - [u(1)]_L$ is identically zero, and the numerical dissipation term becomes

$$d^* = \frac{\alpha \mid a \mid}{2}h^n\frac{K_u(n)}{n!}\begin{cases} u^{(n)}(\xi_L) - u^{(n)}(\xi_R) & \text{if } n \text{ is even,} \\ u^{(n)}(\xi_L) + u^{(n)}(\xi_R) & \text{if } n \text{ is odd.} \end{cases} \tag{32}$$

By Taylor expansion around the location of the interface, $u^{(n)}(\xi_L)$ and $u^{(n)}(\xi_R)$ can be expressed as

$$u^{(n)}(\xi_L) = [u^{(n)}(1)]_L - \theta_L h[u^{(n+1)}(1)]_L + \mathcal{O}(h^2), \tag{33}$$

$$u^{(n)}(\xi_R) = [u^{(n)}(-1)]_R + \theta_R h[u^{(n+1)}(-1)]_R + \mathcal{O}(h^2), \tag{34}$$

where θ_L and θ_R are two numbers in the interval $[0:n]$, such that $\xi_R = \theta_R h - 1$ and $\xi_L = 1 - \theta_L h$. Moreover, for sufficiently smooth solutions, $[u^{(n)}(1)]_L = [u^{(n)}(-1)]_R = u_0^{(n)}$, and $[u^{(n+1)}(1)]_L = [u^{(n+1)}(-1)]_R = u_0^{(n+1)}$. Therefore by using Eq. (34) and retaining leading terms only in the Taylor expansions, Eq. (32) can be further simplified as

$$d^* = \frac{\alpha \mid a \mid}{2}h^n\frac{K_u(n)}{n!}\begin{cases} -(\theta_L + \theta_R)hu_0^{(n+1)} + \mathcal{O}(h^2) & \text{if } n \text{ is even,} \\ 2u_0^{(n)} + \mathcal{O}(h) & \text{if } n \text{ is odd.} \end{cases} \tag{35}$$

From Eq. (35), two main observations are made: (a) when n is an even number, the numerical dissipation term is of the order of h^{n+1}, whereas for odd values of n, the numerical dissipation is of the order of h^n; (b) whatever the order of the scheme, the numerical dissipation term is always proportional to some odd-order derivative of the solution. The first point represents one of the main results of this study: the numerical dissipation term for the whole family of energy stable FR schemes with an even number of collocation points is one order of magnitude smaller than in the case an odd number of points is used. In other words, for the same number of DoF, hence for the same overall resolution, odd orders are expected to be a bit more dissipative than even orders. In any case, the overall dissipation operator in each element, which is obtained from the difference of right and left interface contributions, is proportional to an (undivided) high derivative of even order of the solution, thus providing more efficient absorption of the energy of the unresolved modes [19]. In the next section, the above results will be generalized to the case of Gauss–Legendre quadrature points.

4. The numerical dissipation for Gauss–Legendre solution points

When the solution points are the zeros of a Legendre polynomial, the term $\omega_n(\xi)$ in Eq. (21) represents, by definition, a Legendre polynomial of degree n with unitary leading coefficient. It is worthwhile recalling that by Rodrigues' formula, the Legendre polynomial of degree k can be written as

$$L_k(\xi) = \frac{1}{2^k k!}\frac{\mathrm{d}^{(k)}}{\mathrm{d}\xi^{(k)}}\left[\left(\xi^2 - 1\right)^k\right], \tag{36}$$

hence the corresponding coefficient of the leading monomial term is equal to $a_k = (2k)!/[2^k(k!)^2]$. In evaluating $\omega_n(\xi)$, this coefficient is carried along as a factor. Furthermore, Legendre polynomials in the interval $[-1:1]$ satisfy the conditions

$$L_k(-1) = (-1)^k, \quad \text{and} \quad L_k(1) = 1. \tag{37}$$

Therefore, after rescaling over an element of size nh, the values of $\omega_n(\xi)$ at the interfaces become

$$\omega_n(\xi)|_{\xi=1} = \left(\frac{nh}{2}\right)^n\frac{2^n(n!)^2}{(2n)!}L_n(1) = K_L(n)h^n, \tag{38}$$

$$\omega_n(\xi)|_{\xi=-1} = \left(\frac{nh}{2}\right)^n\frac{2^n(n!)^2}{(2n)!}L_n(-1) = K_L(n)(-h)^n, \tag{39}$$

with

$$K_L(n) = \frac{n^n(n!)^2}{(2n)!}. \tag{40}$$

Due to the similarities between Eqs. (38) and (39) and Eqs. (22) and (23), analogous reasoning as in the previous section leads to the dissipative flux term, which in this case is equal to

Table 1
Values of $K_u(n)/n!, K_L(n)/n!$ and c (*cf.* Eq. (20)) up to order 8.

n	$K_u(n)/n!$	$K_L(n)/n!$	c
2	3.750×10^{-1}	3.333×10^{-1}	3.333×10^{-1}
3	3.125×10^{-1}	2.250×10^{-1}	2.963×10^{-2}
4	2.734×10^{-1}	1.524×10^{-1}	9.524×10^{-4}
5	2.461×10^{-1}	1.033×10^{-1}	1.612×10^{-5}
6	2.256×10^{-1}	7.013×10^{-2}	1.697×10^{-7}
7	2.095×10^{-1}	4.761×10^{-2}	1.220×10^{-9}
8	1.964×10^{-1}	3.233×10^{-2}	6.389×10^{-12}

$$d^* = \frac{\alpha \mid a \mid}{2} h^n \frac{K_L(n)}{n!} \begin{cases} -(\theta_L + \theta_R) h u_0^{(n+1)} + \mathcal{O}(h^2) & \text{if } n \text{ is even}, \\ 2u_0^{(n)} + \mathcal{O}(h) & \text{if } n \text{ is odd}, \end{cases} \tag{41}$$

thus proving that similar conclusions about the numerical dissipation hold in the case of Gauss–Legendre solution points. Some values of the factors $K_u(n)/n!$ and $K_L(n)/n!$ are reported in Table 1 for different orders n. Notice that both factors are decreasing functions of $n, K_L(n)/n!$ showing a slightly faster decay.

5. Numerical tests

In the present section, the anticipated scaling for the dissipative flux d^* is verified numerically. The numerical platform adopted is a one-dimensional solver based on the energy stable FR method [24]. Tests are performed setting the c parameter of the FR method such as to recover the SD type scheme adopted by Jameson [14]. Accordingly, the solution points are the zeroes of Legendre polynomials and the dissipative flux is expected to scale according to Eq. (41). The solution is initialized over a periodic computational domain of length L using a gaussian profile,

$$u_0(x) = \exp\left[1 - \frac{(x - L/2)^2}{(0.1L)^2}\right], \quad \text{with} \quad x \in [0 : L]. \tag{42}$$

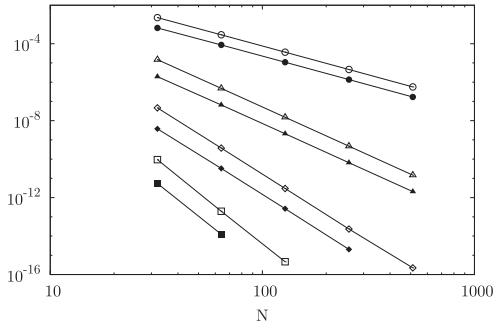
Hence the average dissipative flux is computed for different polynomial orders and for different numbers of elements N as

$$\overline{d^*} = \left[\frac{1}{N}\sum_{i=1}^{N} (u_R - u_L)_i^2\right]^{1/2}, \tag{43}$$

where the index i refers to the elements' interfaces.

The relevant results are represented in Fig. 2, where the initial average dissipative fluxes are plotted with logarithmic scales for N ranging between 32 and 512 and for $n \in [2 : 9]$. The plot clearly shows how the dissipative flux from even orders scales approximately as the dissipative flux of the following odd orders. The quantitative evaluation of the relevant scaling orders, which is reported on the right of Fig. 2, proves that when n is an even (resp. odd) number, the numerical dissipation term is of order of $n + 1$ (resp. n), as anticipated in Section 4.

Fig. 3 shows the time evolution of the average dissipative fluxes (a) and the relevant scaling (b) after the equation is integrated in time for two periods (only half period is plotted to better show the initial behavior). Time integration was done explicitly with a fourth-order Runge–Kutta method (RK4), having care to keep the time step small enough for the relevant error to have a negligible impact on the measured scalings. For analogous reasons, tests were performed for n less than 4, such as to ensure that the discretization error was dominant over the time integration error (read comment below). Although the measured initial dissipative flux follows perfectly the analytical predicted scaling, its time evolution is characterized by a short initial transient followed by an extremely well defined asymptotic value. As it is shown in Fig. 3(b), the scaling of the final asymptotic values of the average dissipative



$n = 2$	-2.99271
$n = 3$	-2.98187
$n = 4$	-4.98712
$n = 5$	-4.96647
$n = 6$	-6.92434
$n = 7$	-6.9393
$n = 8$	-8.81791
$n = 9$	-8.78117

Fig. 2. Dissipation term scaling as a function of the number of elements N: ∘, $n = 2$ (open), $n = 3$ (solid); △, $n = 4$ (open), $n = 5$ (solid); ◇, $n = 6$ (open), $n = 7$ (solid); □, $n = 8$ (open), $n = 9$ (solid). The measured slopes are listed on the right.

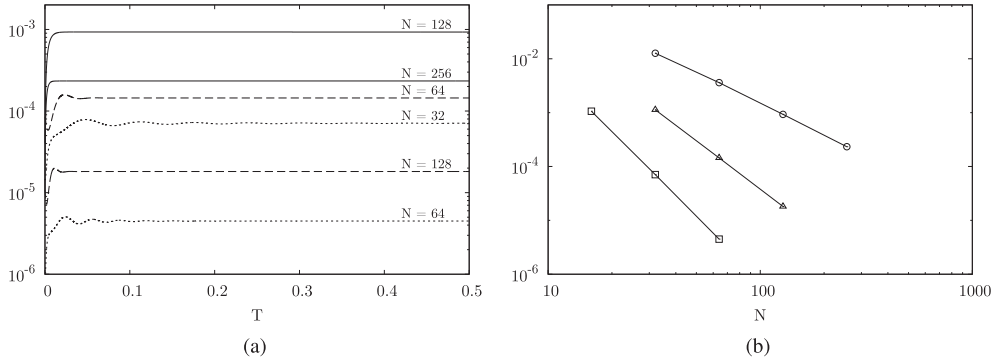
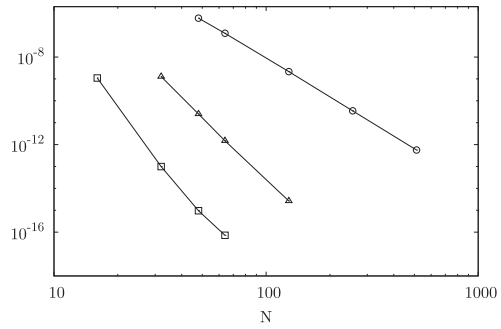


Fig. 3. Time evolution of the average dissipative fluxes (a) and scaling of the relevant asymptotic value (b). In plot (a): ——, $n = 2$; - - - - -, $n = 3$; · · · · · ·, $n = 4$. In plot (b): the measured slopes are −1.92657, −2.98101 and −3.94368 for $n = 2$ (∘), 3 (△) and 4 (□), respectively.



$$\begin{array}{l|r} n=2 & -5.87155 \\ n=3 & -9.4329 \\ n=4 & -12.7831 \end{array}$$

Fig. 4. Scaling of the rate of change of the pth derivative term in the Sobolev energy norm (divided by N): ○, $n=2$; △, $n=3$; □, $n=4$. The measured slopes are listed on the right.

fluxes does not follow the theoretical prediction, but rather reflects the expected order of accuracy of the scheme—namely, n, as anticipated in Section 1—regardless of n being even or odd. Hence, for the linear advection case, apart from a relatively short initial transient, the anticipated difference in behavior between even and odd orders does not seem to have any major impact on the numerical accuracy, for which even and odd orders have similar behavior.

Furthermore, by virtue of Eq. (19), the Sobolev type energy norm of the solution is supposed to decrease in time following the same scaling observed for the dissipative fluxes. More precisely, in the case that periodic boundaries are used (*i.e.*, the second term on the RHS of Eq. (19) is identically zero), from Eqs. (19) and (43) it is expected that the rate of change of the energy norm exhibits the same order scaling of $N \times (\overline{d^*})^2$, *i.e.*,

$$\frac{1}{N}\frac{\mathrm{d}}{\mathrm{d}t}\int_0^L \left[u^2 + \beta^{2p} c u^{(p)^2}\right] \mathrm{d}x \sim 2 \times \mathcal{O}(\overline{d^*}). \tag{44}$$

The two terms in the energy norm, namely, the kinetic energy term u^2 and the one involving the pth derivative of the solution are analyzed below.

With regards to the latter, in particular, from Eqs. (14) and (15), it can be easily shown that the (piecewise constant) pth derivative of the polynomial solution is

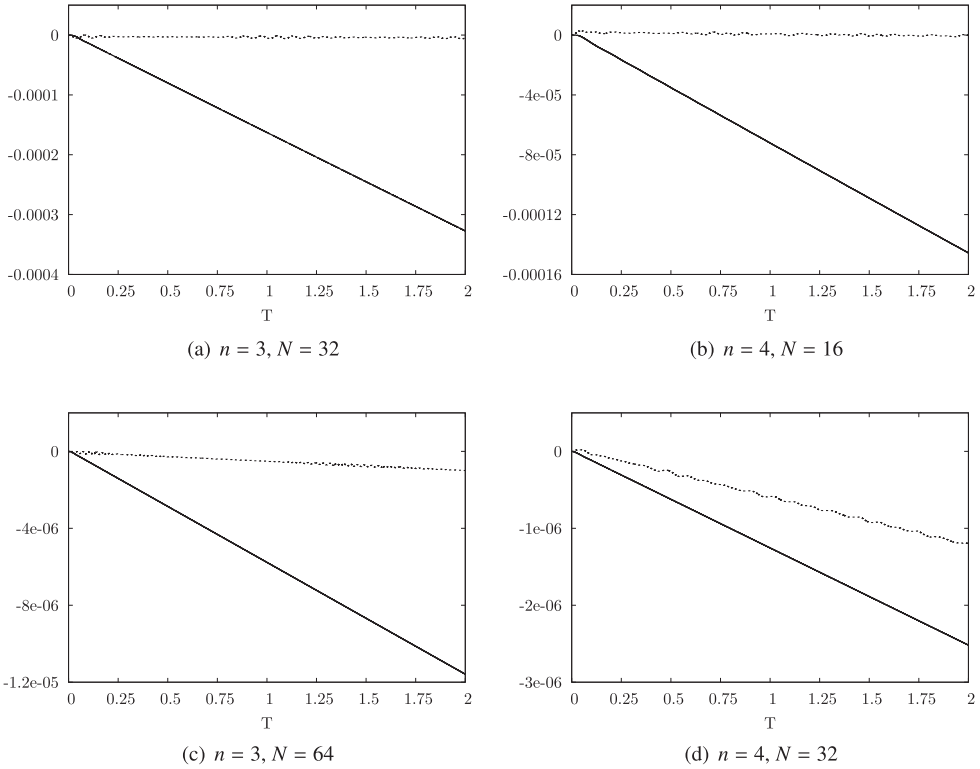
$$\hat{u}^{(p)} = \sum_{i=1}^{n} \hat{u}_i \frac{\mathrm{d}^{(p)} h_i}{\mathrm{d}\xi^{(p)}}, \quad \text{with} \quad \frac{\mathrm{d}^{(p)} h_i}{\mathrm{d}\xi^{(p)}} = p! \prod_{\substack{s=1 \\ s\neq i}}^{n} \frac{1}{\xi_i - \xi_s}, \tag{45}$$

which can be transformed in physical space after rescaling over an element of size nh, namely,

$$u^{(p)} = \frac{(p!)^2 2^{2p}}{(nh)^{2p}} \left[\sum_{i=1}^{n} \hat{u}_i \prod_{\substack{s=1 \\ s\neq i}}^{n} \frac{1}{\xi_i - \xi_s}\right]^2, \tag{46}$$

with $p = n-1$. Recalling that β from Eq. (19) is equal to $(x_{i+1} - x_i)/2 = nh/2$, the pth derivative term in the Sobolev energy norm finally becomes

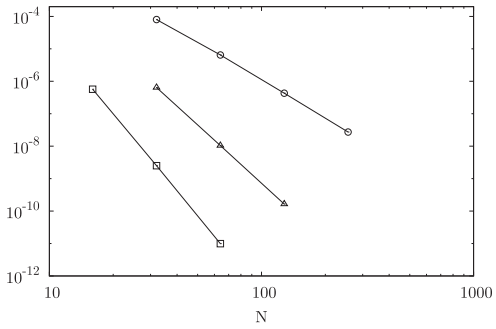
$$\beta^{2p} c u^{(p)^2} = c(p!)^2 \left[\sum_{i=1}^{n} \hat{u}_i \prod_{\substack{s=1 \\ s\neq i}}^{n} \frac{1}{\xi_i - \xi_s}\right]^2, \tag{47}$$



(a) $n=3$, $N=32$ (b) $n=4$, $N=16$

(c) $n=3$, $N=64$ (d) $n=4$, $N=32$

Fig. 5. Time evolution of the normalized error in total kinetic energy at different orders n and for different numbers of elements N: —, with upwind term; ·····, without upwind term.

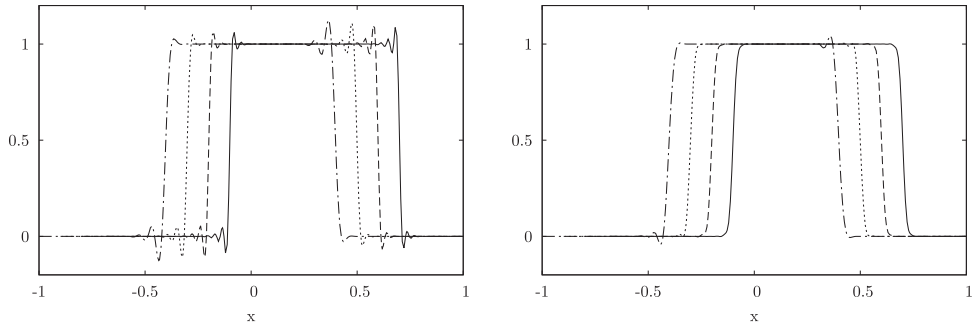


$n = 2$	-3.84411
$n = 3$	-5.96602
$n = 4$	-7.82705

Fig. 6. Scaling of the rate of change of the total kinetic energy m_{FR} (divided by N): ○, $n = 2$; △, $n = 3$; □, $n = 4$. The measured slopes are listed on the right.

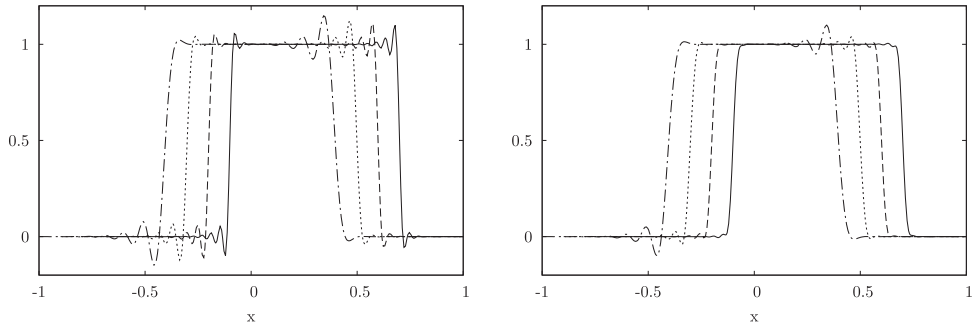
where the coefficient c is obtained from Eq. (20). The scaling of the relevant time derivative—divided by N according to Eq. (44)—as a function of the number of elements N is reported in Fig. 4. As it can be seen, the time derivatives of these terms exhibits a relatively fast decay of the order of N^{-3n} (*cf.* Fig. 4 right). Moreover, their values, which fall below 10^{-6}, are extremely small compared to the corresponding average dissipative fluxes. Therefore, their contribution in Eq. (44) can be neglected. Note that the slight bend observed for $n = 4$ is probably due to uncertainties in the evaluation of slopes of the order of 10^{-16} and smaller.

Turning to the kinetic energy term, since the average dissipative fluxes are almost constant in time (this was observed for sufficiently resolved computations as those shown in Fig. 3), by virtue of Eq. (44), this is expected to decrease linearly in time. This is readily verified in Fig. 5, where the time history of the normalized error in total kinetic energy—namely



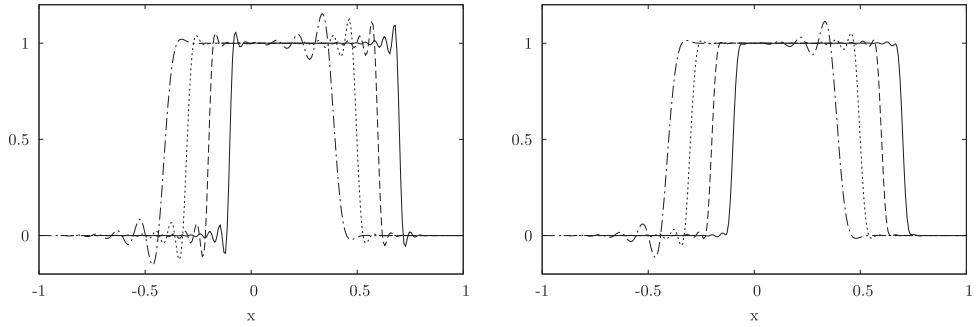
(a) sharp steps after 1 period

(b) smooth steps after 1 period



(c) sharp steps after 10 periods

(d) smooth steps after 10 periods



(e) sharp steps after 20 periods

(f) smooth steps after 20 periods

Fig. 7. Advected sharp (a and c) and smooth (b and d) steps: ——, $n = 6$; - - - - -, $n = 5$; · · · · · ·, $n = 4$; —— · ——, $n = 3$. (curves are shifted along the abscissa to enhance clarity).

$$\mathcal{E}_{\mathrm{ke}} = \frac{\int_0^L (u^2 - u_0^2)\mathrm{d}x}{\int_0^L u_0^2 \mathrm{d}x}, \tag{48}$$

with u_0 from Eq. (42)—is shown for computations of order 3 and 4. Note how, increasing the order n and/or the grid resolution, the numerical dissipation from the time integration scheme tends to become dominant over the effect of the dissipative fluxes. For $n = 5$ and RK4 (not shown) the rate of change of kinetic energy from computations performed with and without dissipative fluxes were indistinguishable, the numerical diffusion from the RK4 being dominant over the spatial discretization counterpart.

The fully centered preliminary computations, in particular, can be used to establish a baseline numerical solution and to better isolate the effect of spatial discretization on the decrease of kinetic energy. In fact, the relevant kinetic energy (also approximately linear in time), which is supposedly only affected by the numerical dissipation introduced by the time integration scheme, allows to perform the decomposition

$$\int_0^L (u^2 - u_0^2)\mathrm{d}x \leqslant I_{\mathrm{FR}}(t) + I_{\mathrm{RK}}(t) \simeq -m_{\mathrm{FR}} t - m_{\mathrm{RK}} t, \tag{49}$$

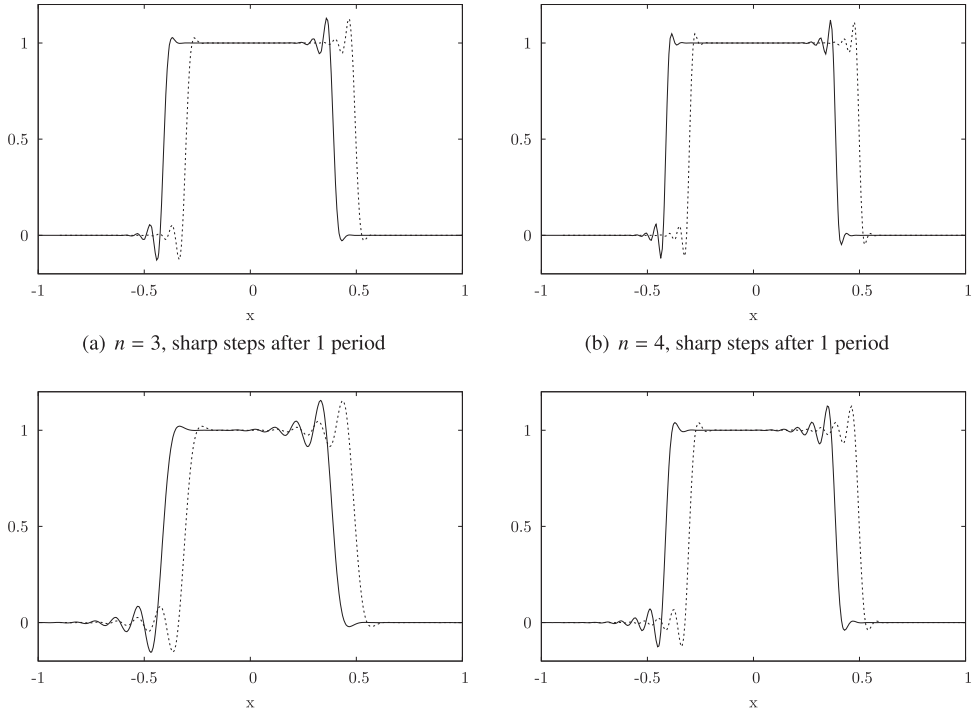
where $I_{\mathrm{FR}}(t)$ and $I_{\mathrm{RK}}(t)$ represent the (time dependent and approximately linear) contributions from the spatial and temporal discretizations, respectively. The relevant linear approximations, expressed by the slopes m_{FR} and m_{RK}, have been evaluated by linear regression of the actual data. Hence, the rate of change due to the RK4, namely, m_{RK}, has been subtracted from the slope measured with the dissipative fluxes switched on, such as to ensure the correct evaluation of the relevant scaling. The actual scaling of the rate of change of kinetic energy m_{FR} is plotted in Fig. 6. Compared to the second term in the Sobolev energy norm (see Fig. 4), the total kinetic energy is of leading order and, as expected, the same scalings observed for the (asymptotic) average dissipative fluxes are fully recovered (note that, according to Eq. (44), the slopes in Fig. 6 have to be twice the slopes in Fig. 3b).

In order to further check for different dissipative behaviors of even and odd order polynomial reconstructions, tests are performed involving the advection of step functions as reported by Bouche et al. [1] in the case of standard finite-difference schemes. The relevant results are plotted in Fig. 7 for sharp and smooth step functions (note that the curves obtained for different values of n are shifted along the abscissa to enhance clarity). Several computations have been performed at constant number of DoF (*i.e.*, $n \times N = \mathrm{cnst.}$) for n ranging from 3 to 6. Note that the computational domain is periodic and that the sharp step involves the initialization of the flow with two Heaviside step functions where the change in value occurs over a single solution point, whereas, in the case of smooth steps, the two Heaviside step functions are smeared along an interval $\Delta = 12L/(n \times N)$ using hyperbolic tangent functions:

$$u(x) = \frac{1}{2}\left[1 + \tanh\left(4\frac{0.4 \pm x}{\Delta}\right)\right], \tag{50}$$

where the plus sign within the argument of the hyperbolic tangent holds for $x \leqslant 0$ and the minus sign otherwise. The above value of Δ, which is designed to be the same regardless of n, was purposely selected by trial and error to produce an almost oscillation free solution when running with $n = 6$.

In contrast with the results reported for standard finite-difference schemes [1], in the case of sharp initial step functions (*cf.* Figs. 7a, c and e), similar short range oscillatory behavior is observed regardless of the order of accuracy n. This confirms previous observations, according to which, the theoretically predicted different amount of numerical dissipation from even and odd orders is not observed in practice in the long time behavior of the solution. Although all the curves are characterized by weakly damped short range oscillations on both sides of the step functions, stronger oscillations are observed upwind of the step functions and their



(a) $n = 3$, sharp steps after 1 period

(b) $n = 4$, sharp steps after 1 period

(c) $n = 3$, sharp steps after 20 periods

(d) $n = 4$, sharp steps after 20 periods

Fig. 8. Advected sharp steps: —, initial discontinuity location inside the element; · · · · · ·, initial discontinuity location at the element interface. (curves are shifted along the abscissa to enhance clarity).

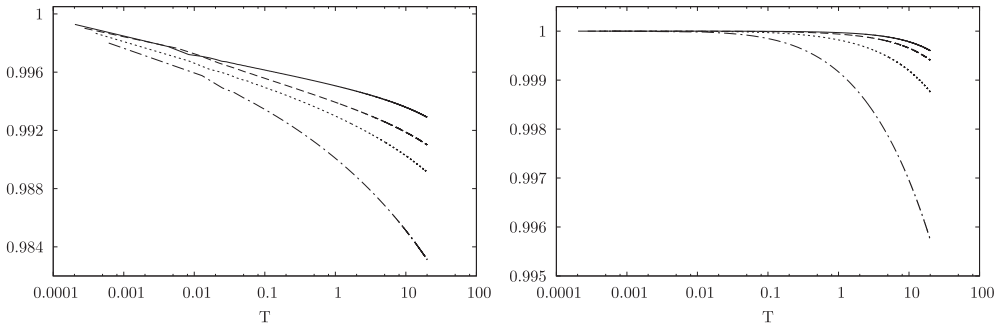


Fig. 9. Time history of the normalized total kinetic energy from the advected sharp (a) and smooth (b) steps: —, $n = 6$; -----, $n = 5$; ·····, $n = 4$; — · —, $n = 3$.

amplitude appears weakened when the order is increased due to the improved resolving power of the scheme (*cf.* discussion below).

Note that the tests reported in Fig. 7 have been performed centering the initial sharp steps such as to locate the discontinuity at the element interface. Analogous tests performed having care to set the initial location of the discontinuity inside the element demonstrate that the solution does not exhibit any significant dependency on the initial location of the discontinuity. The solutions obtained initializing the discontinuity inside the element or at the element interface (curves from Figs. 7) are compared in Fig. 8 for $n = 3$ and 4 ($n = 5$ and 6 exhibit an identical behavior and are omitted).

When a smoothed step function is adopted (see Fig. 7b, d and f), even if the geometrical resolution of the grid is kept constant (*i.e.*, constant number of DoF), the computation performed with the highest value of n produces improved results with very small fluctuations around the (smoothed) step functions. It is worthwhile noting that the initial smoothed profile has been designed to be completely independent of the grid (*viz.* N) and the order (*viz.* n), therefore, regardless of the value of n, the number of solution points across the step is almost fixed. As a result, the observed improvement in the solution can be exclusively attributed to the increased resolving power of the scheme, which is the only parameter that changes across the different computations. As it is shown in Fig. 9, the total kinetic energy of the solution is monotonically decreasing for both the sharp and smooth step functions. This provides further evidence of stability in the energy norm. Notice that similar results (not shown) have been obtained with the c parameter set to zero such to recover the nodal DG scheme [24]. Compared to the presented results, oscillations where in this case somewhat reduced and predominantly downwind of the step functions.

6. Conclusions

The source of numerical dissipation which is embedded in the relatively broad class of energy stable FR schemes has been thoroughly evaluated from both the analytical and numerical points of view.

Starting from the error estimate accompanying the polynomial reconstruction of the solution within each element, the analytical form of the dissipative fluxes—namely, the upwind terms of the interface fluxes—has been derived. The relevant scaling as a function of the mesh resolution shows a peculiar "asymmetry" when the order of the adopted polynomial reconstruction is, respectively, even or odd. In particular, it has been proven that the theoretical scaling of numerical dissipation term produced by even and odd polynomial reconstructions shows different behaviors. At the same time, it has been shown that the numerical dissipation term is consistently proportional to an approximation of an odd-order derivative of the solution, which provides efficient selection of high-frequency modes.

Numerical tests performed on the most simple case of the linear advection equation, fully support the anticipated scaling, at least at the beginning of the simulation. Nonetheless, after the solution is advanced in time, the numerical error of the discretization scheme is quickly propagated throughout the computational domain, such that the theoretical scaling of the dissipative fluxes is lost and a different asymptotic scaling is recovered. More precisely, although the theoretical scaling of the dissipative flux accompanying polynomial reconstructions of odd order p is expected to be equal to $p + 2$ (instead of $p + 1$ as in the case of even orders), numerical tests have given evidence that the relevant asymptotic scaling is $p + 1$ regardless of the order p being even or odd.

Regarding the Sobolev type energy norm under which the scheme is provable stable, it is shown that the term proportional the pth derivative of the polynomial solution is negligible compared to the kinetic energy term. Accordingly, identical scaling has been observed for the rate of decrease of the total kinetic energy and the sum, extended over the whole domain, of the dissipative fluxes, thus confirming that the interface upwind terms represent the main source of numerical damping (the other main source comes from the numerical error introduced by the time integration scheme). Hence, although the analytical form of the upwind terms suggests that computations involving polynomials of odd order are characterized by a somewhat weaker numerical dissipation, the evolution of the upwind terms themselves, in the case of linear advection, appears to be dominated by the numerical error, thus reflecting the nominal order of accuracy of the scheme. This notwithstanding, one might conjecture that the reduced numerical damping of high-frequency modes yielded by odd order computations might tend to produce more marked effects in the nonlinear case, where the energy can be "transversally" redistributed across different frequencies. Future work will be devoted to further analyze potential differences in terms of the numerical dissipation produced by even and odd order polynomial reconstructions in the non-linear case and in more realistic applications involving the integration of the Navier–Stokes equations (*e.g.*, freely decaying homogeneous isotropic turbulence).

Acknowledgements

Financial support under NSF Grant No. 0915006 monitored by Dr. Leland Jameson, and AFOSR Grant No. FA 9550-07-1-0195 from the Computational Math Program under the direction of Dr. Fariba Fahroo, is gratefully acknowledged. The second author was partly supported by CNRS under the INSA Turbulence and Simulation Chair.

Appendix A. The error from interpolating polynomials

Consider n points $x_1, \ldots, x_n$ and a function $f(x) \in \mathbb{C}^n$ in the interval $[x_1 : x_n]$. The remainder of the interpolating polynomial $P_{n-1}(x)$ is then

$$f(x) - P_{n-1}(x) = \frac{(x-x_1)(x-x_2)\cdots(x-x_n)}{n!} f^{(n)}(\xi), \tag{A.1}$$

where $f^{(n)}$ is the nth derivative of f, and $\xi \in [x_1 : x_n]$. The proof of this equation can be found in textbooks (see for instance Ref. [13]) and is here reported for the sake of completeness.

Let $S_n(x)$ be defined by

$$f(x) - P_{n-1}(x) = \omega_n(x) S_n(x), \tag{A.2}$$

where $\omega_n(x) = (x-x_1)(x-x_2)\cdots(x-x_n)$. The function

$$F(z) = f(z) - P_{n-1}(z) - \omega_n(z) S_n(x), \tag{A.3}$$

is then continuous in z and vanishes at $n+1$ points $x_1, \ldots, x_n, x$. Therefore, by Rolle's theorem, $F'(z)$ vanishes at n points, $F''(z)$ vanishes at $n-1$ points, ..., and finally $F^{(n)}(z)$ vanishes at one point ξ. Since

$$\frac{\mathrm{d}^{(n)}}{\mathrm{d}z^{(n)}} P_{n-1}(z) = 0, \quad \text{and} \quad \frac{\mathrm{d}^{(n)}}{\mathrm{d}z^{(n)}} \omega_n(z) S_n(x) = n! S_n(x), \tag{A.4}$$

then

$$F^{(n)}(z) = f^{(n)}(z) - n! S_n(x), \tag{A.5}$$

and setting $z = \xi$ and using Eq. (A.2)

$$S_n(x) = \frac{1}{n!} f^{(n)}(\xi), \tag{A.6}$$

$$\Rightarrow f(x) - P_{n-1}(x) = \frac{\omega_n(x)}{n!} f^{(n)}(\xi) \quad \square \tag{A.7}$$

References

[1] Bouche D, Bonnaud G, Ramos D. Comparison of numerical schemes for solving the advection equation. Appl Math Lett 2003;16(2):147–54.
[2] Castonguay P, Vincent P, Jameson A. A new class of high-order energy stable flux reconstruction schemes for triangular elements. J Sci Comput 2012;51(1):224–56.
[3] Cockburn B, Hou S, Shu C. The Runge–Kutta local projection discontinuous Galerkin finite element method for conservation laws. IV: The multidimensional case. Math Comput 1990;54(190):545–81.
[4] Cockburn B, Lin S, Shu C. TVB Runge–Kutta local projection discontinuous Galerkin finite element method for conservation laws. III: One-dimensional systems. J Comput Phys 1989;84(1):90–113.
[5] Cockburn B, Shu C. TVB Runge–Kutta local projection discontinuous Galerkin finite element method for conservation laws. II: General framework. Math Comput 1989;52:411–35.
[6] Cockburn B, Shu C. The local discontinuous Galerkin finite element method for convection–diffusion systems. SIAM J Numer Anal 1998;35:2440–63.
[7] Cockburn B, Shu C. The Runge–Kutta discontinuous Galerkin finite element method for conservation laws. V: Multidimensional systems. J Comput Phys 1998;141:199–224.
[8] Cockburn B, Shu C. Runge–Kutta discontinuous Galerkin methods for convection-dominated problems. J Sci Comput 2001;16:173–261.
[9] Harten A. High resolution schemes for hyperbolic conservation laws. J Comput Phys 1983;49(3):357–93.
[10] Hesthaven JS, Warburton T. Nodal discontinuous galerkin methods: algorithms, analysis, and applications. Springer Science+Business Media, LLC; 2008.
[11] Huynh H. A flux reconstruction approach to high-order schemes including discontinuous Galerkin methods. In: 18th AIAA computational fluid dynamics conference. Miami, FL, June 25–28, 2007; 2007. AIAA p. 2007–4079, 1–42.
[12] Huynh H. A reconstruction approach to high-order schemes including discontinuous Galerkin for diffusion. In: 47th AIAA aerospace sciences meeting, Orlando, FL, January 5–8, 2009; 2009. AIAA p. 2009–403, 1–34.
[13] Isaacson E, Keller HB. Analysis of numerical methods. John Wiley & Sons, Inc; 1966.
[14] Jameson A. A proof of the stability of the spectral difference method for all orders of accuracy. J Sci Comput 2010;45(1):348–58.
[15] Jameson A. Advances in bringing high-order methods to practical applications in computational fluid dynamics. In: 20th AIAA computational fluid dynamics conference, Honolulu, Hawaii, June 27–30, 2011; 2011. AIAA p. 2011–3226, 1–27.
[16] Jameson A, Vincent P, Castonguay P. On the non-linear stability of flux reconstruction schemes. J Sci Comput 2012;50(2):434–45.
[17] Kopriva D, Kolias J. A conservative staggered-grid Chebyshev multidomain method for compressible flows. J Comput Phys 1996;125(1):244–61.
[18] Liu Y, Vinokur M, Wang Z. Spectral difference method for unstructured grids. I: Basic formulation. J Comput Phys 2006;216(2):780–801.
[19] Mattsson K, Svärd M, Nordström J. Stable and accurate artificial dissipation. J Sci Comput 2004;21(1):57–79.
[20] Roe P. Approximate Riemann solvers, parameter vectors, and difference schemes. J Comput Phys 1981;43:357–72.
[21] Rusanov V. The calculation of the interaction of non-stationary shock waves and obstacles. USSR Comput Math Math Phys 1962;1(2):304–20. English translation of the original paper in Vychislitel'noi Mathematicheskoi Fiziki 1961; 1(2): 267–79.
[22] Sun Y, Wang Z, Liu Y. High-order multidomain spectral difference method for the Navier–Stokes equations on unstructured hexahedral grids. Commun Comput Phys 2007;2(2):310–33.
[23] Toro EF. Riemann solvers and numerical methods for fluid dynamics: a practical introduction. 3rd ed. Berlin Heidelberg: Springer-Verlag; 2009.
[24] Vincent P, Castonguay P, Jameson A. A new class of high-order energy stable flux reconstruction schemes. J Sci Comput 2010;47(1):1–23.
[25] Vincent P, Castonguay P, Jameson A. Insights from von Neumann analysis of high-order flux reconstruction schemes. J Comput Phys 2011;230(22):8134–54.
[26] Vincent P, Jameson A. Facilitating the adoption of unstructured high-order methods amongst a wider community of fluid dynamicists. Math Model Nat Phenom 2011;6(3):97–140.
[27] Wang Z, Liu Y, May G, Jameson A. Spectral difference method for unstructured grids. II: Extension to the Euler equations. J Sci Comput 2007;32(1):45–71.
[28] Williams D, Castonguay P, Vincent P, Jameson A. An extension of energy stable flux reconstruction to unsteady, non-linear, viscous problems on mixed grids. In: 20th AIAA computational fluid dynamics conference, Honolulu, Hawaii, June 27–30, 2011; 2011. AIAA p. 2011–3405, 1–37.