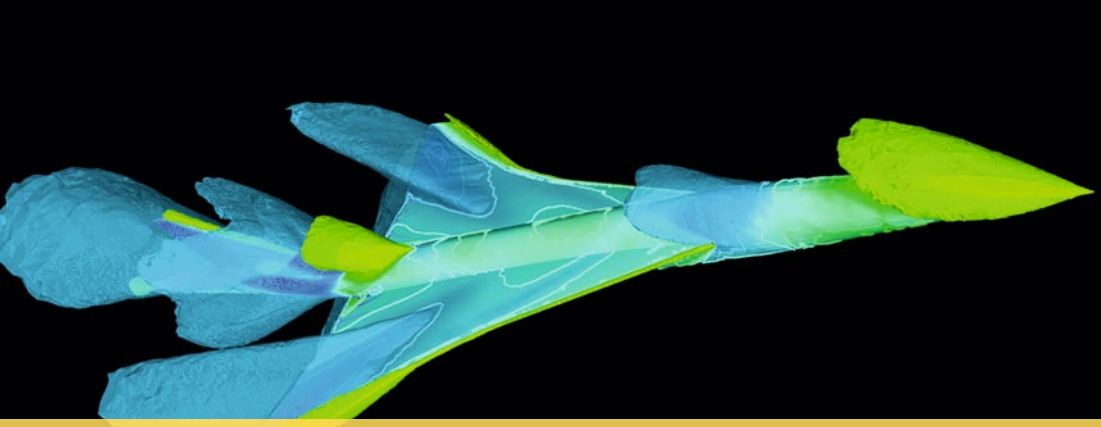


LECTURE NOTES IN COMPUTATIONAL SCIENCE AND ENGINEERING 99

Rémi Abgrall · Héloïse Beaugendre
Pietro Marco Congedo · Cécile Dobrzynski
Vincent Perrier · Mario Ricchiuto *Editors*

High Order Nonlinear Numerical Schemes for Evolutionary PDEs

Springer

Lecture Notes
in Computational Science
and Engineering

99

Editors:

For further volumes:
http://www.springer.com/series/3527

Rémi Abgrall • Héloïse Beaugendre •
Pietro Marco Congedo • Cécile Dobrzynski •
Vincent Perrier • Mario Ricchiuto
Editors

High Order Nonlinear Numerical Schemes for Evolutionary PDEs

Proceedings of the European Workshop HONOM 2013, Bordeaux, France, March 18-22, 2013

Springer

Editors
Rémi Abgrall
Institute of Mathematics
University of Zurich
Zurich, Switzerland

Pietro Marco Congedo
Mario Ricchiuto
Institut de Mathématiques de Bordeaux
INRIA
Talence, France

Héloïse Beaugendre
Cécile Dobrzynski
Institut de Mathématiques de Bordeaux
INRIA and Institut Polytechnique de Bordeaux
Talence, France

Vincent Perrier
Laboratoire de Mathématiques et de leurs Applications
INRIA
Pau, France

ISSN 1439-7358 ISSN 2197-7100 (electronic)
ISBN 978-3-319-05454-4 ISBN 978-3-319-05455-1 (eBook)
DOI 10.1007/978-3-319-05455-1
Springer Cham Heidelberg New York Dordrecht London

Library of Congress Control Number: 2014939848

Mathematics Subject Classification (2010): 35L03, 35L62, 65M08, 65M60, 76M10, 76M12, 76M35

Cover illustration: computation made with a mesh given by F. Alauzet (INRIA)

Printed on acid-free paper

Springer is part of Springer Science+Business Media (www.springer.com)

Preface

This volume gathers contributions presented during the European Workshop on High Order Nonlinear Numerical Methods for Evolutionary PDEs (HONOM 2013) that took place at INRIA Bordeaux Sud-Ouest, Talence, France, from March 18 to 22, 2013. As the title indicates, the theme of this conference was high order methods for evolutionary problems, such as in fluid mechanics, magneto-hydrodynamics, nonlinear solid mechanics and other settings for which genuinely nonlinear methods are needed. Greater emphasis is placed on the numerical than the theoretical aspects of this scientific field: algorithm design, accuracy, large scale computing, complex geometries, and so on.

The 2013 conference was the fifth in the HONOM series. The first was held in Trento, Italy, in 2005, inspired by an idea of Tito Toro, Michael Dumbser and Claus-Dieter Munz. Since then, HONOM has convened every 2 years in Trento, except in 2013 when it was held in Talence near Bordeaux. These conferences, organized under the auspices of the European Congress on Computational Methods in Applied Sciences and Engineering (ECCOMAS), are built on two principles. First they are free to participants, second there is only one session. Free admission is intended to attract as many young participants as possible, and having only one session helps to facilitate the sharing of ideas.

HONOM 2013 featured eight plenary lectures on topics like meshing issue, fractional temporal derivatives for wave propagation, high order schemes, multiphase flows and Vlasov equations. The talks were given by S. Karni (U. of Michigan at Ann Arbor, Michigan, USA), M. Feistauer (Charles University, Prague), J. F. Remacle (Université Catholique de Louvain, Belgium), J. L. Guermond (Texas A&M, College Station, Texas, USA), A. Iske (University of Hamburg, Germany), D. Kopriva (Florida State U., Florida, USA), R. Saurel (Université d'Aix-Marseille, Marseille, France) and E. Sonnendrücker (Max Planck Institute, Garching, Germany). The conference featured some 50 speakers. More than 70 participants attended the sessions among whom only a small fraction came from France.

We would like to thank the following sponsors. First, the French Institute for Research in Computer Science and Automation (INRIA), which hosted the conference and saw to all the practical details. Next, our thanks to GAMNI, the

University of Bordeaux, the Commissariat à l'Énergie Atomique (Le Barp Center), Wiley, Deutsche Forschunggemeinschaft under SPP Metström and SFB-TRR-75, La Communauté urbaine de Bordeaux and the European Research Council (ERC) which has funded the Advanced Research Grant ADDECCO, contract # 226316 as well as the ERC starting grant project StiMulUs, contract #278267.

The next HONOM conference will be held in Trento, Italy, from March 16 to 20, 2015.

Zurich, Switzerland — Rémi Abgrall
Talence, France — Héloïse Beaugendre
Talence, France — Pietro Marco Congedo
Talence, France — Cécile Dobrzynski
Pau, France — Vincent Perrier
Talence, France — Mario Ricchiuto

Contents

Chapter 1
High–Order Asymptotic–Preserving Methods for Nonlinear Relaxation from Hyperbolic Systems to Convection–Diffusion Equations

Sebastiano Boscarino and Giovanni Russo

Abstract In Boscarino et al. (SIAM J Sci Comput, 6(2), A377–A395, preprint: arxiv.org/pdf/1210.4761) the authors propose an asymptotic–preserving method based on Implicit-Explicit (IMEX) Runge-Kutta (R-K) schemes which are adopted to deal with a class of nonlinear hyperbolic systems containing nonlinear diffusive relaxation. These schemes are able to solve such systems with no stiff nor parabolic restriction on the time step. In the limit when the relaxation parameter vanishes, the proposed scheme relaxes to a semi-implicit scheme for the limit nonlinear diffusion equation, thus overcoming the classical parabolic CFL condition in the time step. In this paper we consider an extension of the numerical treatment of the Kawashima-LeFloch's model (LeFloch and Kawashima, private communication) proposed in Boscarino et al. (SIAM J Sci Comput, 6(2), A377–A395, preprint: arxiv.org/pdf/1210.4761). We show that the same schemes introduced in Boscarino et al. (SIAM J Sci Comput, 6(2), A377–A395, preprint: arxiv.org/pdf/1210.4761) relaxes to an semi-implicit scheme for the limit nonlinear convection-diffusion equation. A numerical example confirms the robustness and the accuracy of the scheme in order to capture the correct behavior of the solution in the hyperbolic–to–parabolic regime.

1.1 Introduction

We consider nonlinear hyperbolic systems with stiff nonlinear relaxation. In the parabolic scaling, when the stiffness parameter ε vanishes, the dynamics is asymptotically governed by effective systems which are of parabolic type and may contain possibly degenerate and fully nonlinear diffusion terms.

S. Boscarino • G. Russo (✉)
Dipartimento di Matematica e Informatica, Università di Catania Viale A. Doria 6, 95125 Catania, Italy
e-mail: boscarino@dmi.unict.it; russo@dmi.unict.it

R. Abgrall et al. (eds.), *High Order Nonlinear Numerical Schemes for Evolutionary PDEs*, Lecture Notes in Computational Science and Engineering 99,
DOI 10.1007/978-3-319-05455-1_1,

Fully nonlinear relaxation terms arise, for instance, in presence of *strong friction*. A commonly used model for 1D shallow water problems with friction discussed, for example in [2] (and the references therein), is given by the following system

$$\begin{aligned} &\varepsilon\, h_t + (hv)_x = 0, \\ &\varepsilon\, (hv)_t + \left(hv^2 + p(h)\right)_x = -\frac{\kappa^2(h)}{\varepsilon^2}\, ghv|hv|, \end{aligned} \tag{1.1}$$

where $h > 0$ denotes the water height and v the velocity, while $p(h) = gh^2/2$ plays the role of the pressure. Moreover, $g > 0$ is the gravity constant, and the friction coefficient $\kappa = \kappa(h)$ is a positive function, a standard choice being $\kappa(h) = \kappa_0/h$ with $\kappa_0 > 0$ [19]. It was shown in [2] that the effective nonlinear diffusion equation associated with (1.1) reads

$$h_t = \left(\frac{\sqrt{h}}{\kappa(h)} \frac{h_x}{\sqrt{|h_x|}} \right)_x . \tag{1.2}$$

Directly motivated by the above model and looking for a general class of models of the same type, LeFloch and Kawashima in (private communication) proposed the following class of *nonlinear relaxation model*

$$\begin{aligned} u_t + v_x &= 0, \\ \varepsilon^2\, v_t + b(u)_x &= -|v|^{m-1}\, v + q(u), \end{aligned} \tag{1.3}$$

where $m > 0$ is a real parameter. This, again, is a strictly hyperbolic system of balance laws, provided $b'(u) > 0$. By formally writing

$$|v|^{m-1}\, v = q(u) - b(u)_x + \text{higher-order terms},$$

computing v and plugging its expression in the u-equation, the effective system associated with the model (1.6) is found to be

$$\begin{aligned} u_t &= \Big(|-q(u) + b(u)_x|^{\alpha} \big(-q(u) + b(u)_x \big) \Big)_x , \\ v &= |q(u) - b(u)_x|^{\alpha} (q(u) - b(u)_x), \end{aligned} \tag{1.4}$$

with $\alpha = -1 + 1/m$.
Like the model of strong friction (cf. (1.2)), the system (1.4) contains a *fully nonlinear parabolic* equation in u while, once u is known, the component v is given explicitly by the second formula. It is natural to distinguish between three rather distinct behaviors:

$$\begin{aligned} \text{sub–linear} &: 0 < m < 1, \\ \text{linear} &: m = 1, \\ \text{super–linear} &: m > 1. \end{aligned} \tag{1.5}$$

Equation (1.4), although parabolic in nature, does not regularize solutions with non smooth initial data, and we may expect jump singularities (in the first–order derivative u_x) to remain in the late–time limit.

In [9] we study and analyze several model problems with linear and nonlinear relaxation, including Kawashima-LeFloch's model (LeFloch and Kawashima, private communication). The analysis is performed on the simplest nonlinear model problem (see [9] for details)

$$\begin{aligned} u_t + v_x &= 0, \\ \varepsilon^2 v_t + u_x &= -|v|^{m-1} v, \end{aligned} \tag{1.6}$$

where $m > 0$ is a real parameter. As the parameter ε vanishes the system reduced to the associated effective equation

$$u_t = \left(|u_x|^{\alpha}(u_x)\right)_x, \quad |v|^{m-1} v = -u_x, \tag{1.7}$$

with $\alpha = -1 + 1/m$.

In order to solve numerically such problems we use a method of line. First we discretize in space by conservative finite difference methods in the sense of Shu [22] and obtain a large system of ODE's. For such system, we introduce implicit–explicit (IMEX) methods which are based on Runge–Kutta (R–K) discretization in time. In general, Implicit–Explicit (IMEX) Runge–Kutta (R–K) schemes [1, 3, 4, 12, 21] are successfully adopted in the numerical solution of stiff systems, in which stiffness does not involve the whole system. In many cases the stiffness is associated to some variables. For example, if a system can be written in the partitioned form

$$\begin{aligned} y' &= f(y, z), \\ \varepsilon z' &= g(y, z), \end{aligned} \tag{1.8}$$

then the stiffness is associated to variable z, and the corresponding equation will be treated implicitly, while the equation for y is treated explicitly. In other cases it is more convenient to associate the stiffness to a part of the right hand-side, for example if a system has the additive form

$$u' = f(u) + \frac{1}{\varepsilon} g(u). \tag{1.9}$$

where $\varepsilon > 0$ is the stiffness parameter. In this case the term f is treated explicitly while $\frac{1}{\varepsilon} g(u)$ is treated implicitly. It can be shown that the same system can be written in either form, however sometimes one of the two forms is more convenient.

Section 1.2 is devoted to a brief introduction on IMEX-Runge-Kutta schemes. More detailed description will be found in several papers on the topic, such as [1, 4, 6, 12, 21].

In [9] we showed that the use of classical IMEX R-K methods will lead to consistent explicit discretization of the limit equation (1.7). Because of this, the limit scheme will suffer from the typical parabolic CFL restriction $\Delta t \propto \Delta x^2$. Furthermore in some cases such restriction does not allow the computation of the

solution due to the unboundedness of the diffusion coefficient (super-linear case). In order to overcome this drawback one can make use of a penalization technique, introduced and explained in Sect. 1.4, (see for details [7, 9, 11]). By this technique in the limit $\varepsilon \to 0$ the IMEX R-K scheme converges to a semi-implicit one for the first equation in (1.7) and (1.4). Then these IMEX R-K schemes are able to solve the hyperbolic system containing nonlinear relaxation without any restriction on the time step.

In order to do this, two kinds of approaches for IMEX R-K scheme are considered in [9]. The first is denoted as *partitioned approach* [5, 10, 11] and the corresponding IMEX R-K scheme is called **IMEX-I** R-K scheme, such scheme is applied to systems written in the partitioned form (1.8). The second approach is denoted *additive* [7, 8] and the corresponding scheme is called **IMEX-E R-K** scheme and is adopted when a system is written in the additive form (1.9). In [9] a very efficient IMEX-I R-K scheme has been introduced, in order to solve the first equation in (1.7). The method proposed was stable, linearly implicit and could be designed up to any order of accuracy. In this paper we consider an interesting extension about the numerical treatment of the Kawashima-LeFloch's model proposed in [9]. We apply the same scheme proposed in [9] to solve numerically the *fully non linear parabolic* equation for u in (1.4), i.e. the limit equation of the system (1.3) and we show that the same schemes introduced in [9] are able to solve hyperbolic systems with non linear diffusive relaxation and in the limit they become semi-implicit schemes for the limit nonlinear convection-diffusion equation. At the end in Sect. 1.5, a numerical example confirms the robustness and the accuracy of the scheme which captures the correct behavior of the solution in the hyperbolic–to–parabolic regime.

1.2 Implicit–Explicit (IMEX) Runge-Kutta (R-K) Schemes

Let us consider a Cauchy problem for a system of ordinary differential equations of the additive form

$$y' = f(y) + g(y), \quad y(0) = y_0, \quad t \in [0, T] \tag{1.10}$$

with $y(t) \in \mathbb{R}^N$, and f, g two Lipschitz continuous functions $\mathbb{R}^N \to \mathbb{R}^N$.

Because Runge-Kutta schemes are one-step methods, it is enough to show how to compute the solution after one time step, say from time 0 to time $\Delta t = h$.

An IMEX R-K scheme has the form

$$Y_i = y_0 + h \sum_{j=1}^{i-1} \tilde{a}_{ij} f(t_0 + \tilde{c}_j h, Y_j) + h \sum_{j=1}^{i} a_{ij} g(t_0 + c_j h, Y_j),$$

$$y_1 = y_0 + h \sum_{i=1}^{s} \tilde{b}_i f(t_0 + \tilde{c}_i h, Y_i) + h \sum_{i=1}^{s} b_i g(t_0 + c_i h, Y_i),$$

where $\tilde{A} = (\tilde{a}_{ij})$, $\tilde{a}_{ij} = 0$, $j \geq i$ and $A = (a_{ij})$ are $s \times s$ lower triangular matrices, while $\tilde{c}, \tilde{b}, c, b \in \mathbb{R}^s$ are s-dimensional vectors. IMEX R-K schemes are usually represented by a Double Butcher *tableau*

$$\begin{array}{c|c} \tilde{c} & \tilde{A} \\ \hline & \tilde{b}^T \end{array} \qquad \begin{array}{c|c} c & A \\ \hline & b^T \end{array}.$$

The fact that the implicit scheme is diagonally implicit (DIRK) makes the implementation of IMEX simpler, and ensures that f is effectively explicitly computed. Order conditions for IMEX R-K schemes can be derived by matching Taylor expansion of exact and numerical solution up to a given order p, just as in the case of standard Runge-Kutta schemes. In addition to standard order conditions on the two schemes, additional *coupling conditions* may appear, whose number increases very quickly with the order of the scheme. However, if $\tilde{c} = c$ and $\tilde{b} = b$, then there are no additional conditions for IMEX-RK up to order $p = 3$. For more details on the order conditions consult [12].

1.3 Computation of the Limit Solution

Equation (1.4) is a nonlinear diffusion equation in u. Here we assume that $b(u) = u$ and $q(u) = \beta u$ with $\beta > 0$, i.e.

$$u_t = \Big(|-\beta u + u_x|^{\alpha} \big(-\beta u + u_x \big) \Big)_x . \tag{1.11}$$

One of the difficulties when solving Eq. (1.11) numerically is that it could be stiff and the natural way is to use a fully implicit method. Unfortunately this equation is nonlinear and a fully implicit method would be very expensive. Now in order to solve such equation, avoiding the difficulties of a fully implicit scheme, we shall resort a semi-implicit treatment. We propose here to use a technique presented in [9]. The idea is the following.

Let us consider a partitioned system of the form

$$\begin{aligned} y' &= f(y, z), \\ \varepsilon z' &= g(y, z). \end{aligned}$$

Introducing the variable $w = \varepsilon z$, the system may be written as

$$\begin{aligned} y' &= f(y, w/\varepsilon), \\ w' &= g(y, w/\varepsilon) \end{aligned}$$

which is a particular case of the system

$$u' = F(u, u/\varepsilon), \tag{1.12}$$

with $F : \mathbb{R}^m \times \mathbb{R}^m \to \mathbb{R}^m$ sufficiently differentiable and $u = (y, w)^T$. It is useful to consider system (1.12), in which the first argument of F is treated explicitly, while the second has to be treated implicitly.

Sometime the stiffness is hidden, and is not explicitly expressed in terms of a small positive parameter ε. For this reason we shall consider system of the form

$$u' = F(u, u), \tag{1.13}$$

in which the dependence on the second argument is stiff. Equation (1.11) is of the form (1.13), where the non linear term $|-\beta u + u_x|^\alpha$ is treated explicitly, while the linear term, which determines the stiffness of the system, is treated implicitly.

The subdivision between stiff and non stiff terms has some degree of freedom. The choice is determined by looking for a simple scheme that satisfied both accuracy and stability requirements. The resulting scheme will be denoted semi-implicit.

The semi-implicit IMEX R-K method applied to (1.13) is implemented as follows. First compute the stage fluxes for $i = 1, \dots, s$

$$U_i^* = u_n + \Delta t \sum_{j=1}^{i-1} \tilde{a}_{ij} K_j, \qquad \bar{U}_i = u_n + \Delta t \sum_{j=1}^{i-1} a_{ij} K_j, \tag{1.14}$$

then solve the equation

$$K_i = F(U_i^*, \bar{U}_i + \Delta t a_{ii} K_i), \tag{1.15}$$

and, finally update the numerical solution

$$u_{n+1} = u_n + \Delta t \sum_{i=1}^{s} b_i K_i. \tag{1.16}$$

In our case the function $F(u^*, u)$ is given by

$$F(u^*, u) = (|-\beta u^* + u_x^*|^\alpha (-\beta u + u_x))_x.$$

It can be shown that such a semi-implicit R-K scheme includes Zhong's additive Runge-Kutta methods [23]. The theory developed in [23] can be extended in a straightforward manner to this semi-implicit R-K methods (1.14–1.16) by setting $F(u, u) = f(u) + g(u)$.

Note that, for the spatial discretization of the limiting Eq. (1.4), we can use a second order centered scheme, such as

$$\left(|-\beta u+u_x|^{\alpha}(-\beta u+u_x)\right)_x \approx \Delta x\left(\left(|-\beta U_{j+\frac{1}{2}}(t)+\delta U_{j+\frac{1}{2}}|^{\alpha}\right)\right.$$
$$\times\left(U_{j+\frac{1}{2}}(t)+\frac{U_{j+1}(t)-U_j(t)}{\Delta x}\right)-\dots$$
$$\left(|-\beta U_{j+\frac{1}{2}}(t)+\delta U_{j+\frac{1}{2}}|^{\alpha}\right)$$
$$\left.\times\left(U_{j-\frac{1}{2}}(t)+\frac{U_j(t)-U_{j-1}(t)}{\Delta x}\right)\right),$$

where $|\delta U|^{\alpha}_{j+\frac{1}{2}}=\left|\frac{U_{j+1}(t)-U_j(t)}{\Delta x}\right|^{\alpha}$ and $U_{j+\frac{1}{2}}=\frac{U_{j+1}(t)+U_j(t)}{2}$.

Observe that for the final numerical solution, under the additional assumption $b_i=\tilde{b}_i\ \forall i$, for the numerical solution we get $u^*_{n+1}=u_{n+1}$. This implies that no duplication of variables is needed and no extra computational cost is required to compute the numerical solution.

Note, however, that even in the general case, for a method which is consistent to order p, one has:

$$u_{n+1}=u_{n+1}+\mathcal{O}(h^p),\quad u^*_{n+1}=u^*_{n+1}+\mathcal{O}(h^p).$$

Assuming that $u^*(t_n)=u(t_n)$, then one has $u^*_{n+1}=u_{n+1}+\mathcal{O}(h^p)$ which means that if we neglect the difference between u^*_{n+1} and u_{n+1} and choose, for example, to advance in time with u_{n+1} and to set $u^*_{n+1}=u_{n+1}$, then one obtains another scheme still of order p, with no duplication of variables. Furthermore if we compute u^*_{n+1} by a lower order scheme, obtained with a difference choose of b_i, then one can construct an embedded scheme, which can be used for an automatic time step control [13].

Remark 1. In this paper we use a partitioned approach by an **IMEX-I** scheme, which results in a simpler formulation. For example, it allows $b_i=\tilde{b}_i$ for all i. An additive approach by an **IMEX-E** scheme is more complicated, because it implies $b_i\neq\tilde{b}_i$, for all i, and requires additional order conditions on the coefficients if $q(u)\neq 0$ in order to achieve the second order in time (see [6] for more details).

Here in order to compute the numerical solution of system (1.13) by (1.14), (1.15) and (1.16) we consider a classical second–order IMEX R–K scheme of type **IMEX-I**, that satisfies the condition $b_i=\tilde{b}_i$ for all i, e.g. SSP-IMEX(3,3,2) scheme introduced in [21]

$$\begin{array}{c|ccc} 0 & 0 & 0 & 0\\ 1 & 1 & 0 & 0\\ 1 & 1/2 & 1/2 & 0\\ \hline & 1/3 & 1/3 & 1/3 \end{array}\qquad \begin{array}{c|ccc} 1/4 & 1/4 & 0 & 0\\ 1/4 & 0 & 1/4 & 0\\ 1 & 1/3 & 1/3 & 1/3\\ \hline & 1/3 & 1/3 & 1/3 \end{array}\ .$$

Similar examples it is possible to propose for other types of IMEX R-K schemes presented in the literature.

1.4 Penalization Techinique

In the hyperbolic–to–parabolic limit in (1.6) and (1.3), the characteristic speeds of the hyperbolic part are of order $1/\varepsilon$ and standard IMEX R-K schemes developed for hyperbolic problems with stiff relaxation [6, 21] are not very efficient, since the standard CFL condition requires $\Delta t = \mathcal{O}(\varepsilon \Delta x)$ which may be prohibitive for small ε. Clearly, in the diffusive regime $\varepsilon \ll \Delta x$, we are led to a too restrictive condition, since it is expected that, for an explicit method, a parabolic–type stability condition $\Delta t = \mathcal{O}(\Delta x^2)$ should suffice.

In the literature most existing works [14, 15, 15–18, 20, 20] on asymptotic preserving schemes for nonlinear hyperbolic system with diffusive relaxation in the limit of infinite stiffness, i.e. $\varepsilon \to 0$ become consistent and *explicit* schemes for the diffusive limit equation and, therefore, suffer from the usual stability restriction $\Delta t = \mathcal{O}(\Delta x^2)$.

Schemes that do avoid the parabolic time–step restriction and provide *fully implicit* solvers (in the case of transport equations) have been analyzed in [11] and [7], where a new formulation of the problem (1.4) (for the case $b(u) = u$ and $q(u) = 0$) was introduced, which was based on the addition of two opposite diffusive terms in the stiff limit. In the limit of infinite stiffness, the resulting scheme is consistent and relaxes to an implicit scheme for the underlying diffusion limit, therefore avoiding the typical parabolic restriction of previous methods.

In the present work, in the context of the fully nonlinear model (1.3), with $q(u) = \beta u$ and $b(u) = u$, we adopt a similar penalization technique proposed in [9] that leads us to high–order asymptotic preserving methods covering the hyperbolic–to–parabolic regime under consideration. A key ingredient is to keep the implicit feature of the scheme with respect to certain components that "degenerate"; for instance, system (1.3) when $m > 1$ has a degenerate diffusion when $-q(u) + b(u)_x$ is small or even vanishes.

In order to remove the parabolic CFL restriction we adopt this penalized technique that consists in adding two opposite terms to the first equation in (1.3), i.e.

$$u_t = -(v + \mu(\varepsilon)(|-\beta u + u_x|^\alpha(-\beta u + u_x)))_x + \mu(\varepsilon)(|-\beta u + u_x|^\alpha(-\beta u + u_x))_x.$$

Here, $\mu(\varepsilon)$ is such that $\mu : \mathbb{R}^+ \to [0,\ 1]$ and $\mu(0) = 1$. When ε is not small there is no reason to add and subtract the term $\mu(\varepsilon)(|-\beta u + u_x|^\alpha(-\beta u + u_x))_x$, therefore $\mu(\varepsilon)$ will be small in such a regime, i.e. $\mu(\varepsilon) \approx 0$ for ε not small.

Now the idea here is to apply the semi-implicit approach proposed in Sect. 1.3. Therefore, we rewrite the previous equation as

$$u_t = -(v + \mu(\varepsilon)\Phi(u^*))_x + \mu(\varepsilon)\Psi(u^*, u)_x, \tag{1.17}$$

with $\Phi(u^*) = (|-\beta u^* + u_x^*|^\alpha(-\beta u^* + u_x^*))$, and $\Psi(u^*, u) = (|-\beta u^* + u_x^*|^\alpha(-\beta u + u_x))$ where afterwards the arguments with the $*$-subscript will be treated explicit. In this case, making use of (1.17), system (1.3) can be written as (1.13)

$$y' = F(y^*, y), \tag{1.18}$$

with $y = \begin{pmatrix} u \\ v \end{pmatrix}$, $y^* = \begin{pmatrix} u^* \\ v^* \end{pmatrix}$, and

$$F(y^*, y) = \begin{pmatrix} -(v^* + \mu(\varepsilon)\Phi(u^*))_x + \mu(\varepsilon)\Phi(u^*, u)_x \\ -u_x + |v|^{m-1}v + \beta u \end{pmatrix}. \tag{1.19}$$

In order to show the validity of this technique for the computation of the solution for the system (1.18), we compare the profiles of the numerical solution for the u variable of the limit equation (1.13) and the one computed from Eq. (1.17) by (1.18). We use the classical second order IMEX R-K SSP(3,3,2) scheme with $b_i = \tilde{b}_i$ introduced in Sect. 1.3, in order to compute the numerical solutions.

1.5 Numerical Test

For our numerical test we integrate the system up to two different final times $T = 1$ and $T = 1.6$. We use periodic boundary conditions in $x \in [-\pi, \pi]$ with initial conditions $u(x, 0) = \cos(x)$ and $v(x, 0) = \sin(x)$ and $\varepsilon = 10^{-4}$. We set two values of m here, namely $m = 1/2$ ($\alpha = 1$) and $m = 2$ ($\alpha = -1$). The profiles of the solutions computed with $N = 96$ points are reported in Figs. 1.1 and 1.2.

The following remarks are in order.

- In the evaluation of the limit solution we use the SSP(3,3,2) scheme and the time step has been chosen to satisfy $\Delta t = C\Delta x$ with $C = 0.25$. Similarly for the solution of the system (1.17) we have chosen $\Delta t = C\Delta x$ with $C = 0.25$. As final time we have $T = 1$. This confirms that in the limit case $\varepsilon \to 0$ the proposed approach relaxes to an semi-implicit scheme for the limit convection-diffusion, and the profile is in perfect agreement with the limit solution (see Figs. 1.1 and 1.2)
- Observe that by integrating for a longer time system (1.17), with a hyperbolic CFL, i.e. $\Delta t_H = C_H \Delta x$ for the case $m = 2$ the numerical solution evolves without any instabilities (see Fig. 1.3) and achieves the limit solution in a few time steps. Instability appears when we integrate the limit equation (1.4) by the explicit method with a parabolic CFL restriction i.e. $\Delta t_D = C_D \Delta x^2$ by choosing $C_D = 0.1$. In order to have stability with the explicit scheme we had to reduce $C_D = 0.025$, resulting in a time step more than 200 time smaller than the one used by the semi-implicit approach.

Finally we compute numerical convergence rate where as reference solution we propose again the numerical solution of (1.4) computed by a second order explicit R-K method with $N = 768$ cells and $\Delta t = 0.025\Delta x^2$.

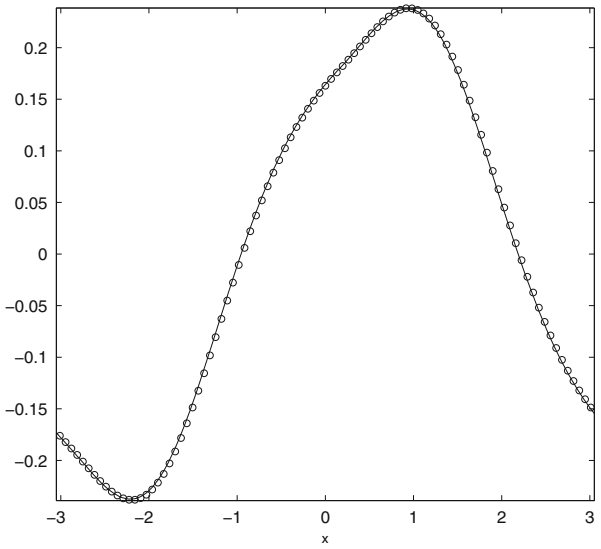


Fig. 1.1 Numerical solution with $N = 96$ cells at time $T = 1$ for $m = 2$. Numerical solution "∘" for system (1.17) with $\Delta t = C\Delta x$, $C = 0.25$ and $\varepsilon = 10^{-4}$. The line is the limit solution of system (1.4) computed bySSP(3,3,2) scheme with $N = 96$, $\Delta t = C\Delta x$, with $C = 0.25$

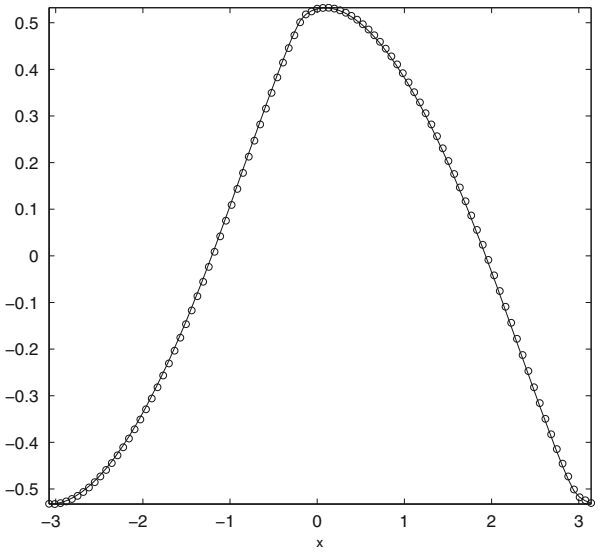


Fig. 1.2 Numerical solution with $N = 96$ cells at time $T = 1$ for $m = 0.5$. Numerical solution "∘" for system (1.17) with $\Delta t = C\Delta x$, $C = 0.25$ and $\varepsilon = 10^{-4}$. The line is the reference solution of system (1.4) computed by SSP(3,3,2) scheme with $N = 96$, $\Delta t = C\Delta x$, with $C = 0.25$

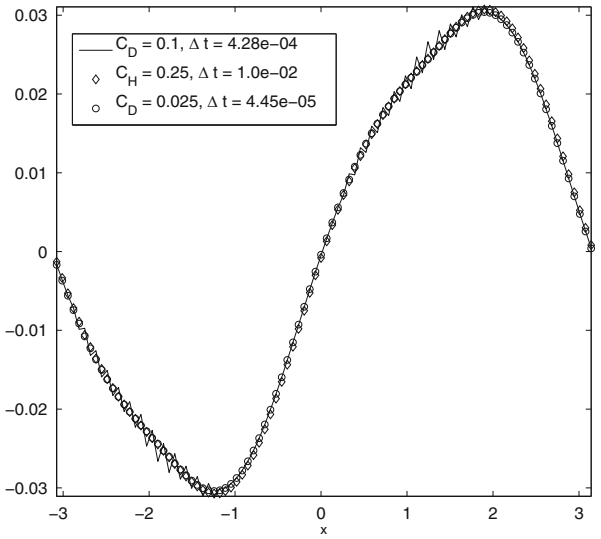


Fig. 1.3 Numerical solutions with $N = 96$ cells at time $T = 1.6$ for $m = 2$. The numerical solutions are compared with different time step: diffusive time step Δt_D and hyperbolic time step Δt_H. The numerical solutions " - ∘ -" and " −" represent the limit solution of system (1.4) computed by an explicit scheme with $\Delta t_D = C_D \Delta x^2$, $C = 0.1$ and $C = 0.025$ respectively. The numerical solution of system (1.17) "◇" with $N = 96$, $\varepsilon = 10^{-4}$ and $\Delta t_H = C_H \Delta x$, and $C = 0.25$

Table 1.1 L^2-norms of the relative error and convergence rates of u with $\Delta t = C\Delta x$, $C = 0.125$ and $m = 2$

N	Error	Order
12	6.0255e − 01	–
24	2.4776e − 01	1.282
48	1.0754e − 01	1.204
96	4.3798e − 02	1.295
192	1.4962e − 02	1.549
384	3.2266e − 03	2.213

We use the formula

$$p = \log_2(E_1/E_2),$$

where E_1 and E_2 are the relative errors as the difference between the reference solution and the numerical one at different time steps using $N = 12 \times 2^p$ with $p = 0, 1, \cdots, 5$. Then one obtains the convergence Table 1.1, the L^2-convergence is second order for $m = 2$, and a CFL hyperbolic condition is used for the time step of the computation of the numerical solution.

Acknowledgements This work has been partially supported by the Italian Project PRIN 2009 "Innovative numerical methods for hyperbolic problems with applications to fluid dynamics, kinetic theory and computational biology".

References

1. Ascher, U., Ruuth, S., Spitheri, R.J.: Implicit-explicit Runge-Kutta methods for time dependent partial differential equations. Appl. Numer. Math. **25**, 151–167 (1997)
2. Berthon, C., LeFloch, P.G., Turpault, R.: Late-time relaxation limits of nonlinear hyperbolic systems. A general framework. Math. Comput. (2012). See also ArXiv:1011.3366
3. Boscarino, S.: Error analysis of IMEX Runge–Kutta methods derived from differential algebraic systems. SIAM J. Numer. Anal. **45**, 1600–1621 (2007)
4. Boscarino, S.: On an accurate third order implicit-explicit Runge-Kutta method for stiff problems. Appl. Numer. Math. **59**, 1515–1528 (2009)
5. Boscarino, S.: Implicit-explicit Runge-Kutta schemes for hyperbolic systems in the diffusion limit. In: ICNAAM 2011, Halkidiki. AIP Conference Proceedings, vol. 1389, pp. 1315–1318 (2011)
6. Boscarino, S., Russo, G.: On a class of uniformly accurate IMEX Runge-Kutta schemes and applications to hyperbolic systems with relaxation. SIAM J. Sci. Comput. **31**(3), 1926–1945 (2009)
7. Boscarino, S., Russo, G.: Flux-explicit ImEx Runge-Rutta schemes for hyperbolic to parabolic relaxation problems. SIAM J. Numer. Anal. **51**(1), 163–190 (2013)
8. Boscarino, S., Russo G.: IMEX Runge-Kutta schemes for hyperbolic systems with diffusive relaxation. In: CD-ROM Proceedings ECCOMAS 2012, Vienna. Vienna University of Technology (2013). ISBN:978-3-9502481-9
9. Boscarino, S., LeFloch, P.G., Russo, G.: High-order asymptotic–preserving methods for fully nonlinear relaxation problems. SIAM J. on Sci. Comput **6**(2), A377–A395. Preprint: arxiv.org/pdf/1210.4761
10. Boscarino, S., Pareschi, L., Russo, G.: IMEX Runge-Kutta schemes and hyperbolic systems of conservation laws with stiff diffusive relaxation, ICNAAM 2009, AIP Conference Proceedings **1168**,1106–1111 (2009)
11. Boscarino, S., Pareschi, L., Russo, G.: Implicit-explicit Runge-Kutta schemes for hyperbolic systems and kinetic equations in the diffusion limit. SIAM J. Sci. Comput. **35**(1), A22–A51 (2013)
12. Carpenter, M.H., Kennedy, C.A.: Additive Runge-Kutta schemes for convection-diffusion-reaction equations. Appl. Numer. Math. **44**(1–2), 139–181 (2003)
13. Hairer, E., Norsett, S.P., Wanner, G.: Solving Ordinary Differential Equation I. Non-stiff Problems. Springer Series in Computational Mathematics, vol. 8, 2nd edn. Springer, Berlin/New York (1993)
14. Jin, S., Pareschi, L.: Discretization of the multiscale semiconductor Boltzmann equation by diffusive relaxation schemes. J. Comput. Phys. **161**, 312–330 (2000)
15. Jin, S., Pareschi, L., Toscani, G.: Diffusive relaxation schemes for multiscale discrete–velocity kinetic equations. SIAM J. Numer. Anal. **35**, 2405–2439 (1998)
16. Klar, A.: An asymptotic-induced scheme for non stationary transport equations in the diffusive limit. SIAM J. Numer. Anal. **35**, 1073–1094 (1998)
17. Lafitte, P., Samaey, G.: Asymptotic–preserving projective integration schemes for kinetic equations in the diffusion limit. SIAM J. Sci. Comput. **34**(2), A579–A602 (2010)
18. Lemou, M., Mieussens, L.: A new asymptotic preserving scheme based on micro–macro formulation for linear kinetic equations in the diffusion limit. SIAM J. Sci. Comput. **31**, 334–368 (2010)

19. Marche, F.: Derivation of a new two-dimensional viscous shallow water model with varying topography, bottom friction and capillary effects. Eur. J. Mech./B-Fluid **26**, 49–63 (2007)
20. Naldi, G., Pareschi, L.: Numerical schemes for hyperbolic systems of conservation laws with stiff diffusive relaxation. SIAM J. Numer. Anal. **37**(4), 1246–1270 (2000)
21. Pareschi, L., Russo, G.: Implicit-explicit Runge-Kutta schemes and applications to hyperbolic systems with relaxations. J. Sci. Comput. **25**(1), 129–155 (2005)
22. Shu, C.W.: Essentially non oscillatory and weighted essentially non oscillatory schemes for hyperbolic conservation laws. In: Advanced Numerical Approximation of Nonlinear Hyperbolic Equations. Lecture Notes in Mathematics, vol. 1697, pp. 325–432. Springer, Berlin/Heidelberg (1998). Series ISSN: 0075-8434
23. Zong X.: Additive semi-implicit Runge-Kutta methods for computing high-speed non equilibrium reactive flows. J. Comput. Phys. **128**, 19–31 (1996)

Chapter 2
A Hybrid Finite Difference-WENO Scheme for Large Eddy Simulation of Compressible Flows

Anne Burbeau-Augoula

Abstract For a wide range of applications, a better understanding of the physical mechanisms involved during the development of hydrodynamic instabilities at interfaces separating two gases is desired. We are interested in flows that contain both turbulence and shock waves, and that are characterized by a high Reynolds number. Current computational power does not allow direct numerical simulation of these flows and then only a large eddy simulation (LES) is possible. This work aims at developing a numerical scheme for compressible LES. Turbulence is better simulated when the numerical method is not dissipative and a high-order of accuracy is recommended for under-resolved simulations. However, the presence of shocks implies the use of a shock-capturing type scheme to stabilize the solution, with the risk of (eventually) overwhelming the small scales of the solution. A compromise can be obtained by coupling a high-order central finite difference scheme for turbulent regions with a WENO scheme for discontinuous ones. To reduce instabilities in non-dissipative LES of turbulent flows, the convective operators are expressed using a skew-symmetric like form. The semi-discretized problem in space should be carefully constructed to ensure the local preservation of kinetic energy by convection. By considering a symmetric stencils collection for the WENO flux reconstruction, the final hybrid flux can be expressed as the sum of the non-dissipative term and a dissipative one. The solution smoothness is estimated component-wise during the WENO reconstruction in the local characteristic space, to avoid the use of excessive numerical dissipation.

A. Burbeau-Augoula (✉)
CEA/DAM Ile-de-France, Bruyères-le-Châtel, 91297 Arpajon Cedex, France
e-mail: anne.burbeau@cea.fr

R. Abgrall et al. (eds.), *High Order Nonlinear Numerical Schemes for Evolutionary PDEs*, Lecture Notes in Computational Science and Engineering 99,
DOI 10.1007/978-3-319-05455-1_2,

2.1 Introduction

In this work, we aim at better understanding hydrodynamic instabilities that develop at interfaces separating two gases, like the Richtmyer-Meshkov (RM) or the Rayleigh-Taylor (RT) instabilities, and particularly the turbulent mixing zone induced by a rapidly growing instability. This is of interest in a wide range of applications, like for example in the implosion of an inertial confinement fusion target. In this process where nuclear fusion reactions are initiated by heating and compressing a capsule that contains a mixture of Deuterium and Tritium, RM instabilities develop when the capsule is shock accelerated.

The turbulent mixing caused by RM instabilities can be investigated by shock tube experiments, which provide reliable quantities such as the mixing zone width and the mean velocity profiles. Besides, numerical simulations give additional informations, in particular concerning the mixing transition and the characterization of the small scale features. In this way, a massively parallel computational code has been developed at CEA for several years to simulate shock tube experiments. It solves the compressible Navier-Stokes equations and a transport equation for the concentration to describe the mixing of two species. It is based on Cartesian geometry. Since current computational power does not allow Direct Numerical Simulation (DNS) of flows under interest for realistic Reynolds numbers, a Large Eddy Simulation (LES) option is under development.

The motivation of this paper is to present a numerical method adapted to LES of compressible flows. The flows we consider contain both shock waves and turbulence, which are very different physical phenomena. The point of view adopted in this work is that only a hybrid approach seems adapted to the contradictory requirements we have to deal with. On the one hand, for the numerical method to be able to capture shocks, it is necessary to introduce numerical dissipation through a shock capturing scheme. On the other hand, the scheme should be non-dissipative in regions of the flow that are smooth, in particular for turbulent flows whose small scales are overwhelmed by any numerical dissipation. For these smooth turbulent solutions, a central non-dissipative scheme should be considered, coupled with a subgrid-scale modelling to represent the effects of the small scales on the resolved (large) scales [5].

A special attention should be paid to the stability property. The fact is that numerical instability is particularly problematic in non-dissipative LES of turbulent flows. Indeed, the cut-off width scale is in between the large scales of the flow and the Kolmogorov length (typically in the inertial zone range where kinetic energy is essentially not dissipated) and consequently the energy is substantial at all the resolved wave numbers. For non-dissipative methods, numerical stability can be based on conservation principles for kinetic energy [3, 4, 11–13].

In the present work, a Finite Difference (FD) scheme of any order of accuracy is considered in regions of smooth solutions. It locally conserves mass, momentum and total energy. The local conservation of kinetic energy by convection is obtained by expressing the convective terms with the skew-symmetric form proposed by

Kennedy and Gruber [11], and by neglecting time step errors. A WENO procedure is implemented for shocks. Following Martín et al. [16] and Hu et al. [8], and unlike the pioneering method of Jiang and Shu [9], a fully downwind stencil is added to the stencil collection during the flux reconstruction. This way, we find the central FD scheme in smooth regions.

The objectives of the present work are twofold. Firstly, the convective split form proposed by Kennedy and Gruber [11] is reviewed, in combination with the locally conservative spatial discretization of Pirozzoli [18]. The reason governing such work is that the fully discrete system of equations is rarely presented (or only partially) and the discretization of the equation for total energy is the key point for ensuring stability. It is shown how to construct FD operators (in the general case of any order of accuracy) for the split form, which satisfy both primal and secondary conservation laws [17]. Unlike the scheme recently proposed by Morinishi [17], our scheme is not fully conservative because the time errors are supposed to be zero. Discrete analogous of continuous operators expressing the convection of a given quantity and of this quantity per square are constructed. The correct discretization of total energy equation is then established.

Secondly, a new hybrid method is investigated in which the solution regularity of the Euler equations is analyzed in the local characteristic space introduced during the WENO flux reconstruction. Then, the artificial dissipation is added component-wise, which results in a less dissipative method.

In the first part, the problem of numerical instabilities is examined, and the finite difference scheme for smooth solutions of non-dissipative equations is detailed. The second part is devoted to the WENO scheme for solutions with discontinuities. Then the hybrid approach, which couples the FD and the WENO schemes by using a shock sensor is presented. Finally, some numerical results are proposed, mainly to illustrate both the robustness of the proposed numerical method and the advantages of using a high-order spatial discretization.

2.2 The Finite-Difference Scheme for Smooth Solutions

2.2.1 On the Problem of Numerical Instability

As previously mentioned, a preliminary work is necessary to treat the problem of numerical instability for non-dissipative LES. The fact is that solutions of the Navier-Stokes equations can develop non-linear instabilities even in the absence of shock waves. These instabilities are essentially attributed to aliasing errors, which arise during the discrete evaluation of the differentiation of the product of two (or more) variables. Several methods exist to prevent aliasing, the most drastic being the "$\frac{2}{3}$ dealiasing rule", which needs to introduce the (very expensive) Fourier transform. To reduce aliasing, a filtering procedure can also be considered, to try to eliminate the high-frequency modes, potentially badly resolved. This kind of

method, inherently dissipative, does not seem to be well adapted to turbulence simulation. At last, the most successful approach consists in reformulating the convective terms in a skew-symmetric like form. Two views coexist to justify this approach. On the one hand, it has been demonstrated that aliasing errors are reduced when considering a relevant split form of the convective operator [11, 13]. On the other hand, the reason for using a skew-symmetric splitting can be the aim to enforce conservation. The global conservation of kinetic energy in the absence of external and viscous forces and time differencing errors insures numerical stability for incompressible flows. For compressible flows, the situation is different since the kinetic energy can increase or decrease due to isentropic exchanges between kinetic energy and internal energy, but we don't want the numerical scheme to produce artificial (non-physical) exchanges between these energies. Experience has shown that, for compressible flows, local conservation of kinetic energy by convection implies better numerical stability.

Below, the skew-symmetric form for the continuous case is presented.

2.2.2 *The Skew-Symmetric Form for the Convective Terms*

The physical model problem is the system of compressible Euler equations. It can be written in divergence form by using the Einstein summation convention as

$$\begin{cases} D\phi \equiv \dfrac{\partial \rho\phi}{\partial t} + \dfrac{\partial}{\partial x_j}\left(\rho u_j \phi\right) = \dfrac{\partial F_{\phi j}}{\partial x_j},\ \phi = \{1, u_i, E\}, i = 1, 2, 3. \\ F_{1,j} = 0,\ F_{u_i,j} = -P\delta_{ij},\ F_{E,j} = -Pu_j. \end{cases} \tag{2.1}$$

In (2.1), ρ stands for the density, u_i for the velocity components, P for the pressure and the total energy per mass unit is denoted by E.

The divergence operator $\dfrac{\partial}{\partial x_j}\left(\rho u_j \phi\right)$ can be rewritten in several analytically equivalent forms. Following Kennedy and Gruber [11], the skew-symmetric form

$$\frac{1}{4}\left(\frac{\partial \rho u_j \phi}{\partial x_j} + u_j \frac{\partial \rho\phi}{\partial x_j} + \rho \frac{\partial u_j \phi}{\partial x_j} + \phi \frac{\partial \rho u_j}{\partial x_j} + \rho u_j \frac{\partial \phi}{\partial x_j} + \rho\phi \frac{\partial u_j}{\partial x_j} + u_j \phi \frac{\partial \rho}{\partial x_j} \right). \tag{2.2}$$

is well adapted to cubically non-linear terms and provides additional robustness for flows with strong density gradients. The compressible flow equations (2.1) can be rewritten as

$$K_P \phi = \frac{\partial F_{\phi j}}{\partial x_j} \tag{2.3}$$

with the definition of the operator

$$K_P\phi \equiv \frac{\partial\rho\phi}{\partial t} + \frac{1}{4}\left(\frac{\partial\rho u_j\phi}{\partial x_j} + u_j\frac{\partial\rho\phi}{\partial x_j} + \rho\frac{\partial u_j\phi}{\partial x_j} + \phi\frac{\partial\rho u_j}{\partial x_j} + \rho u_j\frac{\partial\phi}{\partial x_j} + \rho\phi\frac{\partial u_j}{\partial x_j} + u_j\phi\frac{\partial\rho}{\partial x_j}\right). \tag{2.4}$$

As a prelude to the elaboration of discrete operators, notice that the split form (2.4) has the property of giving both conservation of primal and secondary quantities (according to the terminology of Morinishi [17]). Firstly,

$$K_P\phi = D\phi. \tag{2.5}$$

Consequently, mass, momentum and total energy are conserved when the convective operators are expressed using the split form.

Secondly, we want to ensure that if the convection of a physical quantity (ϕ) is expressed with the convective operator (2.4) then its quadratic form (ϕ^2) will be conserved. Straightforward formal calculus shows that

$$\phi K_P\phi = D\left(\frac{\phi^2}{2}\right) + \frac{1}{2}\phi^2\left(\frac{\partial\rho}{\partial t} + \frac{\partial\rho u_j}{\partial x_j}\right). \tag{2.6}$$

Equation (2.6) states that the preservation of kinetic energy by convection follows immediately if the skew-symmetric form is used to express convection, with the help of the continuity equation. In other words, if we consider that a sufficient condition for stability is the preservation of kinetic energy by convection, then the stability of the form K_P will be supported by a consistency relation with the continuity equation.

The following section is devoted to the FD approximation of the spatial part of the skew-symmetric form. The key is the careful construction of approximate operators which satisfy discrete analogous of Eqs. (2.5) and (2.6).

2.2.3 *The Finite-Difference Approximation of the Skew-Symmetric Form*

Consider a uniform mesh with nodes x_j. Define $x_{j+\frac{1}{2}} = \frac{1}{2}(x_j + x_{j+1})$ and $\Delta x = x_{j+1} - x_j$.

The finite-difference approximation of the first order derivative of a function f at node x_j is given by

$$\begin{aligned}\frac{\partial f}{\partial x}(x_j) \simeq \nabla_j(f) &\equiv \frac{1}{\Delta x}\sum_{l=1}^{L} a_l \left(f(x_j + l\Delta x) - f(x_j - l\Delta x)\right) \\ &\equiv \frac{1}{\Delta x}\sum_{l=1}^{L} a_l \left(f_{j+l} - f_{j-l}\right),\end{aligned} \tag{2.7}$$

where $2L$ stands for the number of points of the stencil. The coefficients a_l can be computed by different methods: in the standard approach, the approximation order for a given value of L is optimized using Taylor's polynomial. Another methods can be considered, like for example the Dispersion-Relation-Preserving method optimization [20]. This is not the topic of current paper in which the standard approach was chosen.

The approximate gradient operator can be expressed in a locally conservative form, suited for compressible flows:

$$\nabla_j f = \frac{1}{\Delta x}\left(\tilde{f}^{2L}_{j+\frac{1}{2}} - \widetilde{f}^{2L}_{j-\frac{1}{2}}\right), \quad \tilde{f}^{2L}_{j+\frac{1}{2}} \equiv \sum_{l=1}^{L} a_l \sum_{m=0}^{l-1}\left(f_{j+l-m} + f_{j-m}\right). \tag{2.8}$$

Starting from the previous definition of the FD operator (2.7), an approximation of the split form (2.4) on cell j is

$$\begin{aligned}(K_P)_j\phi = &\frac{d\rho_j\phi_j}{dt} \\ &+ \frac{1}{\Delta x}\sum_{l=1}^{L} a_l \frac{1}{4}\Big[\ (\rho u\phi)_{j+l} - (\rho u\phi)_{j-l} + u_j\left((\rho\phi)_{j+l} - (\rho\phi)_{j-l}\right) + \\ &\rho_j\left((u\phi)_{j+l} - (u\phi)_{j-l}\right) + \phi_j\left((\rho u)_{j+l} - (\rho u)_{j-l}\right) + \\ &\rho_j u_j\left(\phi_{j+l} - \phi_{j-l}\right) + \rho_j\phi_j\left(u_{j+l} - u_{j-l}\right) + \\ &u_j\phi_j\left(\rho_{j+l} - \rho_{j-l}\right)\Big].\end{aligned}$$

By simply reordering terms we arrive to the formulation

$$\begin{aligned}(K_P)_j\phi = \frac{d\rho_j\phi_j}{dt} + \frac{1}{\Delta x}\sum_{l=1}^{L} a_l \frac{1}{4}\Big[\ &(\rho_j + \rho_{j+l})(u_j + u_{j+l})(\phi_j + \phi_{j+l}) - \\ &(\rho_j + \rho_{j-l})(u_j + u_{j-l})(\phi_j + \phi_{j-l})\Big].\end{aligned} \tag{2.9}$$

Introducing mean average

$$\widetilde{\rho u\phi}_{j,l} = \frac{1}{4}\left[(\rho_j + \rho_{j+l})(u_j + u_{j+l})(\phi_j + \phi_{j+l})\right] \tag{2.10}$$

we get [18]

$$(K_P)_j\phi = \frac{d\rho_j\phi_j}{dt} + \frac{1}{\Delta x}\sum_{l=1}^{L} a_l\left(\widetilde{(\rho u\phi)}_{j,l} - \widetilde{(\rho u\phi)}_{j-l,l}\right). \tag{2.11}$$

At the end, the expected locally conservative form is obtained:

$$(K_P)_j\phi = \frac{d\rho_j\phi_j}{dt} + \frac{1}{\Delta x}\left(H_{j+\frac{1}{2}} - H_{j-\frac{1}{2}}\right), \quad H_{j+\frac{1}{2}} \equiv \sum_{l=1}^{L} a_l \sum_{m=0}^{l-1} \widetilde{(\rho u\phi)}_{j-m,l}. \tag{2.12}$$

This equation stands for a discrete analogous of (2.5).

We now proceed to define a correct discrete analogous for kinetic energy. Let us consider the approximate value of quantity $\phi K\phi$ on cell j:

$$\begin{aligned}\phi_j(K_P)_j\phi = \phi_j\frac{d\rho_j\phi_j}{dt} \\ + \frac{1}{4}\Big(\phi_j\nabla_j(\rho u\phi) + u_j\phi_j\nabla_j(\rho\phi) + \rho_j\phi_j\nabla_j(u\phi) + \\ \phi_j^2\nabla_j(\rho u) + \rho_j u_j\phi_j\nabla_j\phi + \rho_j\phi_j^2\nabla_j u + u_j\phi_j^2\nabla_j\rho\Big).\end{aligned} \tag{2.13}$$

Suppose that (note right now that this relation is not satisfied by the Runge-Kutta scheme employed for the time discrete operator)

$$\frac{d}{dt}\left(\rho_j\frac{\phi_j^2}{2}\right) = \phi_j\frac{d\rho_j\phi_j}{dt} - \frac{1}{2}\phi_j^2\frac{d\rho_j}{dt}. \tag{2.14}$$

Let us introduce the semi-discrete continuity equation (not defined yet)

$$\frac{d\rho_j}{dt} + \tilde{\nabla}_j(\rho u) = 0. \tag{2.15}$$

It yields

$$\begin{aligned}\phi_j(K_P)_j\phi = \frac{d}{dt}\left(\rho_j\frac{\phi_j^2}{2}\right) + \frac{1}{4}\Big(\quad \phi_j\nabla_j(\rho u\phi) + \rho_j u_j\phi_j\nabla_j\phi + \\ u_j\phi_j\nabla_j(\rho\phi) + \rho_j\phi_j\nabla_j(u\phi) + \\ \rho_j\phi_j^2\nabla_j u + u_j\phi_j^2\nabla_j\rho + \phi_j^2\nabla_j(\rho u) - \\ 2\phi_j^2\tilde{\nabla}_j(\rho u)\Big).\end{aligned} \tag{2.16}$$

Some terms of the RHS of (2.16) can be easily put in a locally conservative form. Firstly,

$$\phi_j \nabla_j (\rho u \phi) + \rho_j u_j \phi_j \nabla_j \phi = \frac{1}{\Delta x} \sum_{l=1}^{L} a_l \sum_{m=0}^{l-1} \left[\widetilde{(\phi \rho u \phi)}^{(1)}_{j-m,l} - \widetilde{(\phi \rho u \phi)}^{(1)}_{j-m-1,l} \right] \tag{2.17}$$

with

$$\widetilde{(\phi \rho u \phi)}^{(1)}_{j,l} = \phi_j \phi_{j+l} (\rho_{j+l} u_{j+l} + \rho_j u_j). \tag{2.18}$$

Secondly,

$$u_j \phi_j \nabla_j (\rho \phi) + \rho_j \phi_j \nabla_j (u \phi) = \frac{1}{\Delta x} \sum_{l=1}^{L} a_l \sum_{m=0}^{l-1} \left[\widetilde{(\phi \rho u \phi)}^{(2)}_{j-m,l} - \widetilde{(\phi \rho u \phi)}^{(2)}_{j-m-1,l} \right], \tag{2.19}$$

with

$$\widetilde{(\phi \rho u \phi)}^{(2)}_{j,l} = \phi_j \phi_{j+l} \left(\rho_{j+l} u_j + \rho_j u_{j+l} \right). \tag{2.20}$$

The remaining term

$$\phi_j^2 \left(\rho_j \nabla_j u + u_j \nabla_j \rho + \nabla_j (\rho u) \right) \tag{2.21}$$

cannot be put into a divergence form. To get the correct form, one only has to take benefit from the fact that the continuity equation has not been defined yet. It is interesting to note that the consistency relation between the continuity equation and the equation for kinetic energy previously mentioned in Sect. 2.2.2 comes naturally. Thereby, we can show that the following definition of the discrete continuity equation leads to the cancelling of the term (2.21):

$$\frac{d\rho_j}{dt} + \frac{1}{\Delta x} \sum_{l=1}^{L} a_l \left[\sum_{m=0}^{l-1} \widetilde{(\rho u)}^{(3)}_{j-m,l} - \sum_{m=0}^{l-1} \widetilde{(\rho u)}^{(3)}_{j-m-1,l} \right] = 0 \tag{2.22}$$

with

$$\widetilde{(\rho u)}^{(3)}_{j,l} = \frac{1}{2} (\rho_{j+l} + \rho_j)(u_{j+l} + u_j). \tag{2.23}$$

This completes the elaboration of a locally conservative approximation of $\phi(K_P)\phi$. Indeed, at the end

$$\phi_j (K_P)_j \phi = \frac{d}{dt}\left(\rho_j \frac{\phi_j^2}{2}\right) + \frac{1}{\Delta x}\sum_{l=1}^{L} a_l \left[\sum_{m=0}^{l-1} \widetilde{\widetilde{(\phi\rho u\phi)}}_{j-m,l} - \sum_{m=0}^{l-1} \widetilde{\widetilde{(\phi\rho u\phi)}}_{j-m-1,l}\right] \tag{2.24}$$

with

$$\widetilde{\widetilde{(\phi\rho u\phi)}}_{j,l} = \widetilde{(\phi\rho u\phi)}^{(1)}_{j,l} + \widetilde{(\phi\rho u\phi)}^{(2)}_{j,l} = \frac{1}{4}\phi_j \phi_{j+l} \left(\rho_{j+l} + \rho_j\right)\left(u_{j+l} + u_j\right) \tag{2.25}$$

for the discretization of continuity equation previously given.
Finally, the semi-discretized system of Euler equations has the form

$$\frac{d\phi_j}{dt} + \frac{1}{\Delta x}\left(F^{\phi}_{j+\frac{1}{2}} - F^{\phi}_{j-\frac{1}{2}}\right) = 0 \tag{2.26}$$

with

$$\begin{aligned}
F^{1}_{j+\frac{1}{2}} &= \sum_{l=1}^{L} a_l \sum_{m=0}^{l-1} \frac{1}{2}\left(\rho_{j+l-m} + \rho_{j-m}\right)\left(u_{j+l-m} + u_{j-m}\right), \\
F^{u}_{j+\frac{1}{2}} &= \sum_{l=1}^{L} a_l \sum_{m=0}^{l-1} \frac{1}{4}\left(\rho_{j+l-m} + \rho_{j-m}\right)\left(u_{j+l-m} + u_{j-m}\right)\left(u_{j+l-m} + u_{j-m}\right), \\
F^{E}_{j+\frac{1}{2}} &= \sum_{l=1}^{L} a_l \sum_{m=0}^{l-1} \Bigg[\frac{1}{4} u_{j+l-m} u_{j-m} \left(\rho_{j+l-m} + \rho_{j-m}\right)\left(u_{j+l-m} + u_{j-m}\right) + \\
&\qquad \frac{1}{4}\left(\rho_{j+l-m} + \rho_{j-m}\right)\left(u_{j+l-m} + u_{j-m}\right)\left(e_{j+l-m} + e_{j-m}\right) + \\
&\qquad \frac{1}{2}\left(u_{j+l-m} + u_{j-m}\right)\left(P_{j+l-m} + P_{j-m}\right)\Bigg].
\end{aligned} \tag{2.27}$$

2.3 The WENO Scheme for Shock Waves

The classical WENO scheme [9] can be modified by considering a symmetric stencil collection [8, 16, 21]. Let us briefly recall the method in the context of the one-dimensional conservation law

$$\frac{\partial u}{\partial t} + \frac{\partial f(u)}{\partial x} = 0, \tag{2.28}$$

with positive advection speed $\frac{df}{du} > 0$.

A conservative FD spatial discretization is obtained by introducing a function $h(x)$ implicitly defined as $f(x) = \int_{x-\frac{\Delta x}{2}}^{x+\frac{\Delta x}{2}} h(\xi)d\xi$. The scheme has the form

$$\frac{du_i}{dt} = -\frac{1}{\Delta x}\left[\hat{f}^r_{i+\frac{1}{2}} - \hat{f}^r_{i-\frac{1}{2}}\right], \tag{2.29}$$

where $\hat{f}^r_{i+\frac{1}{2}}$ is a r-th order accurate approximation of $h(x_{i+\frac{1}{2}})$, reconstructed from known cell-average values f_i. It is built by a convex combination of $(r+1)$ candidate numerical fluxes $q^r_{k,i+\frac{1}{2}}$, each calculated from a r-points stencil

$$S_{k,i+\frac{1}{2}} = (x_{i+k-r+1}, \ldots, x_{i+k}). \tag{2.30}$$

It yields

$$\hat{f}^r_{i+\frac{1}{2}} = \sum_{k=0}^{r} \omega_k q^r_{k,i+\frac{1}{2}}, \ \sum_{k=0}^{r} \omega_k = 1. \tag{2.31}$$

To obtain $q^r_{k,i+\frac{1}{2}}$, a polynomial approximation is postulated: $h(x) \simeq \hat{f}_k(x)$, where $\hat{f}_k(x)$ is r-th degree polynomial.

The reason for considering $(r+1)$ candidate stencils (instead of r according to the pioneering method of Jiang and Shu [9]) is that for smooth solutions, the WENO scheme can exactly match the high-order centered FD scheme. Indeed, all candidate stencils all together contain $2r$ grid values of f. It is then possible to construct a $(2r)$-th order approximation of $h_{i+\frac{1}{2}}$, denoted by $\tilde{f}^{2r}_{i+\frac{1}{2}}$. The stencil collection support being symmetric with respect to node $x_{i+\frac{1}{2}}$, the FD scheme introduced in Sect. 2.2.3 should be considered giving

$$\tilde{f}^{2r}_{i+\frac{1}{2}} = \sum_{l=1}^{r} a_l \sum_{m=0}^{l-1} (f_{i+l-m} + f_{i-m}). \tag{2.32}$$

This scheme defined by (2.32) is named the *optimal* central scheme. The corresponding flux can be expressed as a combination of the reconstructed fluxes on each stencil S_k. Simple algebra gives the coefficients d_k such that

$$\tilde{f}^{2r}_{i+\frac{1}{2}} = \sum_{k=0}^{r} d_k q^r_{k,i+\frac{1}{2}}, \ \sum_{k=0}^{r} d_k = 1. \tag{2.33}$$

The weights ω_k are defined such that in smooth regions of the solution, one recover the optimal scheme:

$$\omega_k = \frac{\alpha_k}{\sum_{i=0}^{r} \alpha_i}, \ \alpha_k = \frac{d_k}{(\beta_k + \epsilon)^2}, \tag{2.34}$$

where β_k is the smoothness indicator associated to stencil S_k and ϵ is a small parameter value to avoid division by zero [1, 9]. Here the WENO weights are computed by the classical method:

$$\beta_k = \sum_{j=1}^{r} \Delta x^{2j-1} \int_{x_{j-\frac{1}{2}}}^{x_{j+\frac{1}{2}}} \left(\frac{d^j}{dx_j} \hat{f}_k(x) \right)^2 dx. \tag{2.35}$$

Since the stencil S_r contains only downwind information, it could cause instabilities. Therefore, following [16], after the computation of the smoothness measurements with (2.34), β_r is further constrained to satisfy

$$\beta_r = \max_{0 \le k \le r} \beta_k.$$

To end this part, the WENO flux is the sum of the FD central flux and a dissipative term:

$$\hat{f}^r_{i+\frac{1}{2}} = \tilde{f}^{2r}_{i+\frac{1}{2}} + \sum_{k=0}^{r} (\omega_k - d_k)\, q^r_{k,i+\frac{1}{2}}, \tag{2.36}$$

this way the skew-form based central flux can be considered.

As previously mentioned, for smooth solutions, the dissipative term will tend to zero. However, to reduce as most as possible the dissipation, we want this term to be exactly zero. Then, a hybrid approach should be implemented. This is the topic of the following section.

2.4 The Hybrid Approach

The hybrid approach relies on the coupling of two methods, typically a non-dissipative method for smooth regions and a shock capturing method in regions where the solution is discontinuous. Equivalently, it can be viewed as a centered method with the addition of artificial dissipation as a surrogate of physical viscosity where necessary. The success of such an approach relies on the method employed to detect the discontinuous regions where dissipation might be employed.

Hereafter, the shock-detecting sensor held in this work is first detailed. Then, the hybrid flux is introduced.

2.4.1 A Shock Detecting Sensor

Many methods have been proposed up to now. See for example Li and Qiu [15] for an overview of different detectors. These methods differ from there dependence (or not) on the problem solved. The complexity of the method is also a very important criterion since the main objective of an hybrid approach, besides minimizing the dissipation, is the reduction of the simulation CPU time. The shock-detecting sensor developed by Darian et al. [2] for filtering of high-order compact finite difference schemes is attractive because it is simple to implement and enables an accurate localization of solution discontinuities.

Let f be the function which we want to study the smoothness. An interpolated value of f at the point x_i is computed as

$$\hat{f_i} = f_i - \frac{(-1)^m}{\frac{2m!}{m!m!}} (\Delta\nabla)^m f_i,\ \Delta f_i = f_{i+1} - f_i,\ \nabla f_i = f_i - f_{i-1}. \tag{2.37}$$

with m a positive integer constant (m = 4 in all the computations).

This interpolated value uses the neighbouring points $\{f_{i\pm k}\}_{k=1,\ldots,m}$ and the point value itself f_i is not included in the interpolation. Then, the term $\left|\hat{f_i} - f_i\right|$ is the error of a $2m$-th order interpolation. The following quantity

$$s_i(f) = 1 - \exp\left(-\delta \left|\hat{f_i} - f_i\right|^2\right),\ \delta > 0 \tag{2.38}$$

can act as a smoothness measurement. Indeed, in smooth regions, a Taylor expansion analysis shows that $s_i = \mathcal{O}(\Delta x)^{4m}$, and $s_i = \mathcal{O}(1)$ for a discontinuity. Then, a simple and effective shock-sensor is obtained by introducing a threshold real parameter C and by arguing that a smooth region (respectively a discontinuous region) is detected if $s_i < C$ (respectively $s_i \geq C$). Experience showed that the value of the parameter δ can be taken as $\delta = 100$ for all the simulations. Unfortunately, the parameter C is problem-dependent. For a given numerical simulation, several values are tested to obtain the best compromise between accuracy and stability. In practice, the choice $C \in [0.1, 0.001]$ seems reasonable.

2.4.2 The Hybrid Flux

Once the solution discontinuities are detected, the switch between a dissipative and a non-dissipative method is possible. The usual way to proceed consists in analysing the smoothness of a function which is representative of the problem to be solved and to tag discontinuous regions. This approach is briefly recalled before introducing a new method.

2.4.2.1 The Basic Approach

The solution smoothness is estimated throw the analysis of a given flow variable, typically the density, the entropy function or the vorticity. The shock-detecting sensor $s_i(f)$ is computed to define

$$\eta_{i+\frac{1}{2}} = \begin{cases} 0 \; if \; s_i(f) < C \; and \; s_{i+1}(f) < C \\ 1 \; otherwise. \end{cases} \tag{2.39}$$

The final flux is a linear combination of the central FD flux and the WENO flux:

$$\hat{f}^{Flux,r}_{i+\frac{1}{2}} = \left(1 - \eta_{i+\frac{1}{2}}\right) \tilde{f}^{2r}_{i+\frac{1}{2}} + \eta_{i+\frac{1}{2}} \hat{f}^{r}_{i+\frac{1}{2}}. \tag{2.40}$$

This is the method which is generally considered. One can expect the discontinuity localization to be too few accurate because the solution regularity is estimated throw one variable only.

2.4.2.2 The Proposed Method

To reduce the artificial dissipation, a new method is under study. It is based on the observation that the WENO reconstruction is done in the local characteristic space. It seems then more relevant to apply the shock-detector component-wise, for each characteristic field. Let $\mathbf{l_p}$ be the p-th left eigenvector of the Jacobian. If $s_j\left(\mathbf{l_p} \cdot \mathbf{u}\right) > C$ on at least one stencil S_j of the collection, then we set $(q^r_{k,i+\frac{1}{2}})_p = 0$, otherwise

$$(q^r_{k,i+\frac{1}{2}})_p = q^r_{k,i+\frac{1}{2}}\left(\mathbf{l_p} \cdot \mathbf{f}_{j+k-r+1}, \ldots, \mathbf{l_p} \cdot \mathbf{f}_{j+k}\right). \tag{2.41}$$

The parameter C is problem-dependent.

2.4.3 The Multidimensional Case

For three-Dimensional problems and Cartesian geometries, the system of Euler equations is solved numerically by applying the one-dimensional method previously described in each coordinate direction. As far as the hybrid approach is concerned, when computing the numerical fluxes in a given Cartesian coordinate direction, the solution smoothness is measured in this direction.

2.5 Numerical Results

Three numerical tests are now considered. The first problem of the Taylor-Green vortex flow is aimed at testing the kinetic energy preservation property. Then, the Shu-Osher problem is considered to test the capacity of the hybrid method to capture a shock wave and a complex smooth solution, without producing spurious oscillations characteristics of unstable methods. Finally, a compressible decaying isotropic turbulence simulation illustrates the interest of using the hybrid method rather than the WENO one.

2.5.1 The Taylor-Green Vortex Flow

The Taylor-Green vortex flow is one of the simplest problems to study the breakdown process of large-scale vortices into successively smaller ones. This is a good candidate for testing the accuracy of high-order methods on the direct numerical simulation of a 3D transitional flow and the stability of the method for under-resolved motions. The flow is computed within a periodic square box defined as $0 \leq x_i \leq 2\pi$.

The initial flow field is given by

$$\begin{aligned} \rho &= 1, \\ u_1 &= \sin x_1 \cos x_2 \cos x_3, \\ u_2 &= -\cos x_1 \sin x_2 \cos x_3, \\ u_3 &= 0, \\ P &= 100 + \frac{(\cos(2x_3) + 2)(\cos(2x_1) + \cos(2x_2)) - 2}{16}. \end{aligned} \tag{2.42}$$

This flow transitions to turbulence, with the creation of small scales, followed by a decay phase similar to decaying homogeneous turbulence. The problem is made essentially incompressible by choosing a pressure sufficiently high. Since in the incompressible problem the kinetic energy should remain constant, this is an interesting test for measuring the kinetic energy preservation property.

A Test of Numerical Stability

We consider a very coarse grid to increase the (potential) problem of numerical instability. It contains 32^3 elements.

To investigate the numerical stability property of the proposed method (for a 10-th order of accuracy), it is compared with

1. The 10-th order FD scheme based on the divergence form for all the convective operators (Fig. 2.1),

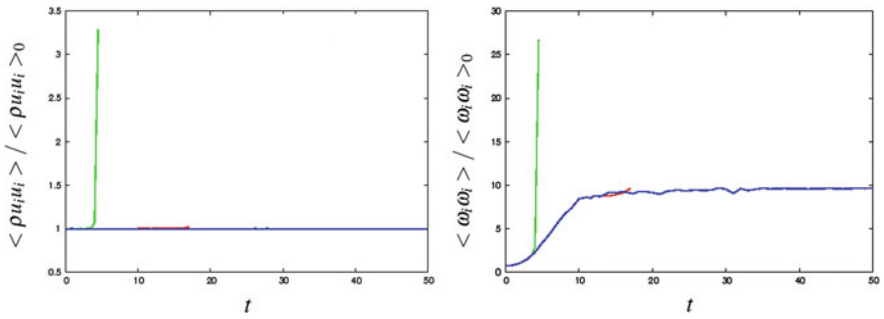


Fig. 2.1 The Taylor-Green vortex flow. Mean quantities on the 64^3 grid. Finite difference scheme. Different formulations for the convective terms: conservative form (*light gray*), skew-form of Honein and Moin (*dark gray*) and skew-form of Kennedy and Gruber (*black*)

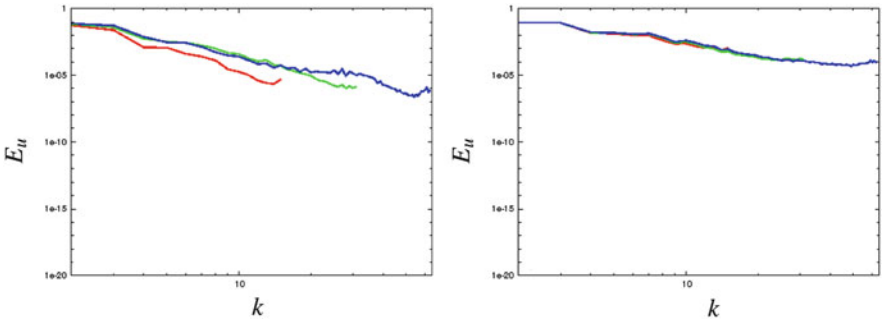


Fig. 2.2 The Taylor-Green vortex flow. Velocity spectra on 32^3 (*dark gray*), 64^3 (*light gray*) and 128^3 (*dark*) grid (logarithmic scales). WENO scheme (*left*) and Hybrid method with C = 0.75 (*right*)

2. The 10-th order FD scheme obtained when expressing the convective terms of momentum and total energy equations with the skew-formulation suggested by Honein and Moin [7], maintaining the divergence form for the continuity equation (Fig. 2.2). It comes to replace

$$\begin{aligned}
\frac{\partial(\rho u_i u_j)}{\partial x_j} &\longrightarrow \frac{1}{2}\frac{\partial(\rho u_i u_j)}{\partial x_j} + \frac{\rho u_j}{2}\frac{\partial u_i}{\partial x_j} + \frac{u_i}{2}\frac{\partial(\rho u_j)}{\partial x_j} \\
\frac{\partial(\rho E + P)u_j}{\partial x_j} &\longrightarrow \frac{1}{2}\frac{\partial}{\partial x_j}\left(\rho e u_j\right) + \frac{e}{2}\frac{\partial \rho u_j}{\partial x_j} + \frac{\rho u_j}{2}\frac{\partial(e)}{\partial x_j} \\
&\quad + P\frac{\partial u_j}{\partial x_j} + u_j\frac{\partial P}{\partial x_j} \\
&\quad + \frac{u_i}{2}\frac{\partial}{\partial x_j}\left(\rho u_i u_j\right) + \frac{\rho u_i u_j}{2}\frac{\partial u_i}{\partial x_j}
\end{aligned} \tag{2.43}$$

In a recent paper, Hill et al. [6] considered this formulation for LES of a Richtmyer-Meshkov instability.

The problem is solved for a long time (t = 50). The computation with the divergence form stopped very early, and that with the formulation of Honein and Moin a little later. The skew-form of Kennedy and Gruber with the scheme developed in this work is only one to reach the end of the simulation.

A Test of Convergence

The velocity spectrum is computed on a sequence of successively refined grids that contain respectively 32^3, 64^3 and 128^3 elements. As mentioned in [10], we can expect the convergence of the spectrum for a fixed time and a limited range of wave-numbers. The WENO scheme is very dissipative on the coarsest grid, and affects all the resolved wave-numbers. The velocity spectra for grids 64^3 and 128^3 seem to converge for $K \leq 16$. The lower wave numbers clearly converge for the hybrid method.

2.5.2 The Shu-Osher Test Case

This is a one-dimensional idealization of shock-turbulence interaction in which a shock propagates into a perturbed density field [19]. The initial flow field is given by

$$(\rho, u, p) = \begin{cases} (3.857143, 2, 629369, 10.33333), & x < -4, \\ (1 + 0.2\sin(5x), 0, 1), & x \geq -4. \end{cases} \tag{2.44}$$

The domain is $x \in [-5, 5]$ with the grid spacing $\Delta x = \dfrac{10}{200}$. The final time is t = 1.8. A reference solution was obtained on a very fine mesh (1,800 elements) with the fifth-order WENO scheme (r = 5). We test the capacity of the method to capture solutions that contain both shocks and complex smooth regions. The influence of the addition of an extra candidate stencil during the WENO reconstruction is first investigated for r = 3 (Fig. 2.3) and r = 5 (Fig. 2.4). The density and entropy profiles show that the inclusion of the downwind stencil gives rise to a less dissipative but stable method.

Then, we compare the results obtained with the hybrid method for two different values of the shock sensor threshold parameter value (C = 0.01 and C = 0.75). The results are plotted in Fig. 2.5 for r = 3 and Fig. 2.6 for r = 5. Good results are obtained with the hybrid method for a correct adjustment of the threshold parameter value. The influence of this parameter value on the numerical solution is less important for the higher-order WENO scheme. The regions where a discontinuity is detected are plotted in Figs. 2.7 (C = 0.01) and 2.8 (C = 0.75). The solution discontinuities are more accurately detected with C = 0.75.

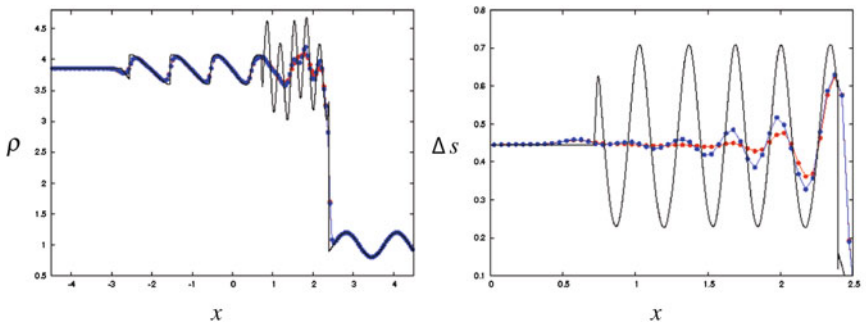


Fig. 2.3 Shu-Osher problem. WENO method for r = 3. Symmetric (*dark symbols*) and non-symmetric (*light gray symbols*) collection of stencils. Reference solution (*black line*). Density profiles (*left*) and zoom of the entropy profile

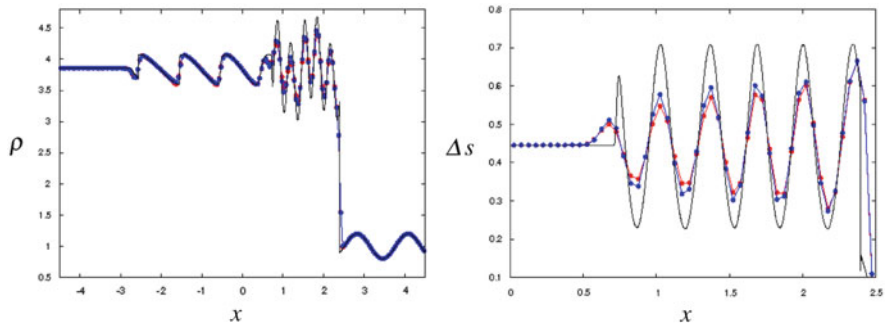


Fig. 2.4 Shu-Osher problem. WENO method for r = 5. Symmetric (*dark symbols*) and non-symmetric (*light gray symbols*) collection of stencils. Reference solution (*black line*). Density profiles (*left*) and zoom of the entropy profile

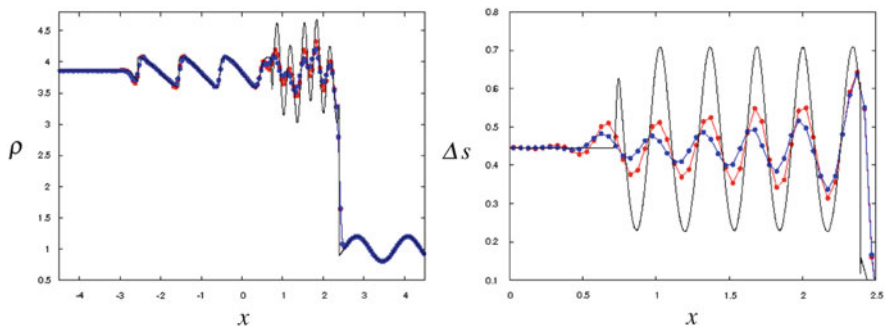


Fig. 2.5 Shu-Osher problem. Hybrid method for r = 3 and two values of the shock-sensor threshold parameter value: C = 0.01 (*dark symbols*) and C = 0.75 (*light gray symbols*). Reference solution (*black line*). Density profiles (*left*) and zoom of the entropy profile

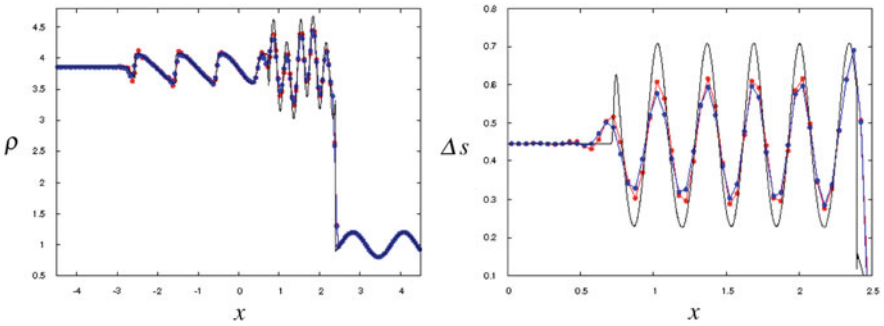


Fig. 2.6 Shu-Osher problem. Hybrid method for r = 5 and two values of the shock-sensor threshold parameter value: C = 0.01 (*dark symbols*) and C = 0.75 (*light gray symbols*). Reference solution (*black line*). Density profiles (*left*) and zoom of the entropy profile

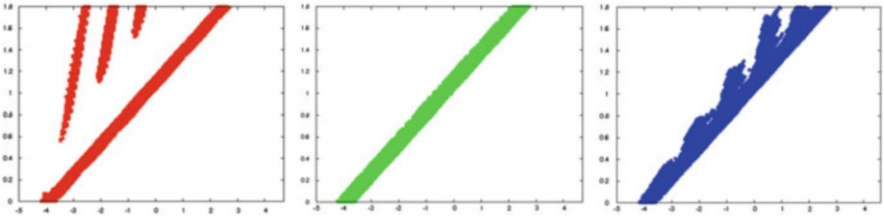


Fig. 2.7 Shu-Osher problem. Hybrid method for r = 5 and C = 0.01. Regions with addition of the dissipative term for (from left to right) the rarefaction wave, the contact discontinuity wave and the shock wave

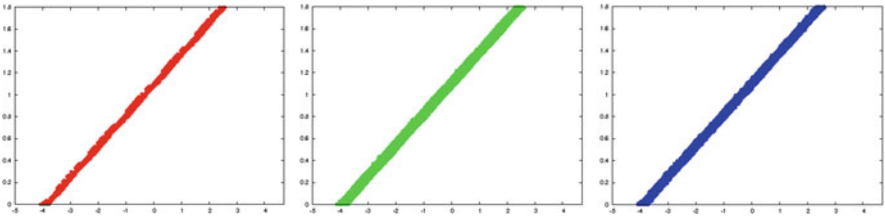


Fig. 2.8 Shu-Osher problem. Hybrid method for r = 5 and C = 0.75. Regions with addition of the dissipative term for the $u - c$ wave, the u wave and the $u + c$ wave

2.5.3 *Compressible Isotropic Turbulence*

The last test is that of decaying compressible turbulence with eddy shocklets [14]. A recent detailed description of the problem can be found in [10], where several numerical methods capable of handling turbulence and shock waves are compared on this problem.

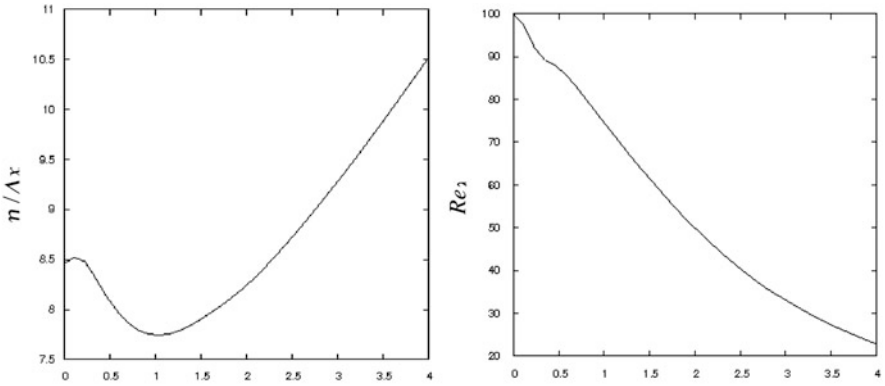


Fig. 2.9 Compressible isotropic turbulence. Time evolution of the Kolmogorov scale and the Reynolds number

We solve the three-dimensional compressible Navier-Stokes equations with $\gamma = 1.4$. The domain is $x_i \in [0, 2\pi]^3$ and the grid spacing $\Delta x_i = \frac{2\pi}{64}$. Periodic boundary conditions are imposed in all directions. The initial condition consists of a random solenoidal velocity field such that $E(k) \sim k^4 \exp\left(-2\left(\frac{k}{k_0}\right)^2\right)$.

The flow can be characterized by two parameters, typically the turbulent Mach number $M_t = \frac{< u_i u_i >}{c}$ and the Reynolds number based on the Taylor micro-scale λ: $Re_\lambda = \frac{< \rho > \lambda \sqrt{\frac{< u_i u_i >}{3}}}{< \mu >}$. The simulation parameters are: $k_0 = 4$, $\lambda = \frac{2}{k_0}$, $M_{t_0} = 0.6$, $Re_\lambda = 100$. The final time of the simulation is $\frac{t}{\tau} = 4$ with $\tau = \frac{\lambda_0}{u_0}$, $u_0^2 = \frac{< \|u\|^2 >}{3}$.

A reference solution was obtained on 256^3 grid. The time evolution of $\eta/\Delta x$ where η is the Kolmogorov scale shows this is a (well resolved) DNS simulation (see Fig. 2.9). The time evolution of the Reynolds number is plotted in the same Figure.

Time evolution of different quantities (Fig. 2.10) and spectra (Fig. 2.11) show that the WENO method under predicts all the results, which is the illustration of excessive dissipation. However, the hybrid method is satisfactory.

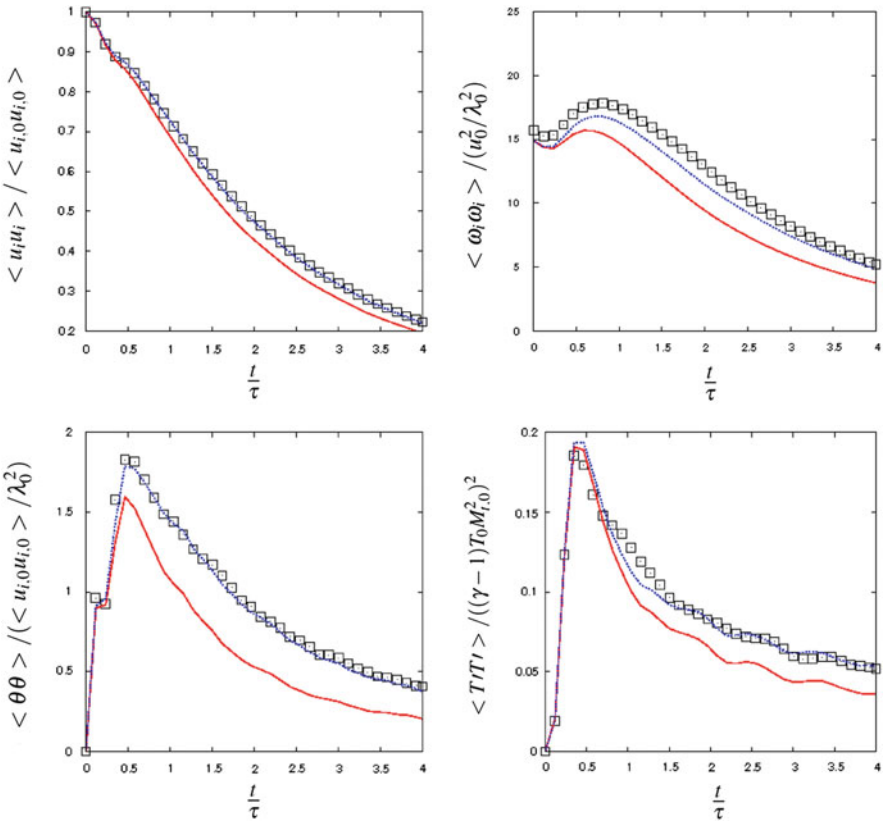


Fig. 2.10 Compressible isotropic turbulence. Time evolution of the variance of different quantities. The reference solution is the solution on a 256^3 grid spectrally filtered to a 64^3 grid (*squares*). WENO (*solid lines*) and Hybrid (*dotted lines*) solutions on 64^3 grid

2.6 Concluding Remarks

In this paper, we have presented a finite difference-WENO method for large eddy simulations of compressible flows. The non-dissipative method is based on the skew-form proposed in [11]. Following [18], we have shown how to construct finite difference schemes of any order of accuracy that preserve kinetic energy by convection. The WENO scheme is a central version of the classical method [9]. This way, the hybrid flux can be expressed as the sum of the central skew-symmetric based flux and the dissipative part of the WENO flux. The smoothness measurement is applied in the local characteristic space to avoid excessive damping of the solution. Future work will be devoted to large eddy simulations of Richtmyer-Meshkov instabilities.

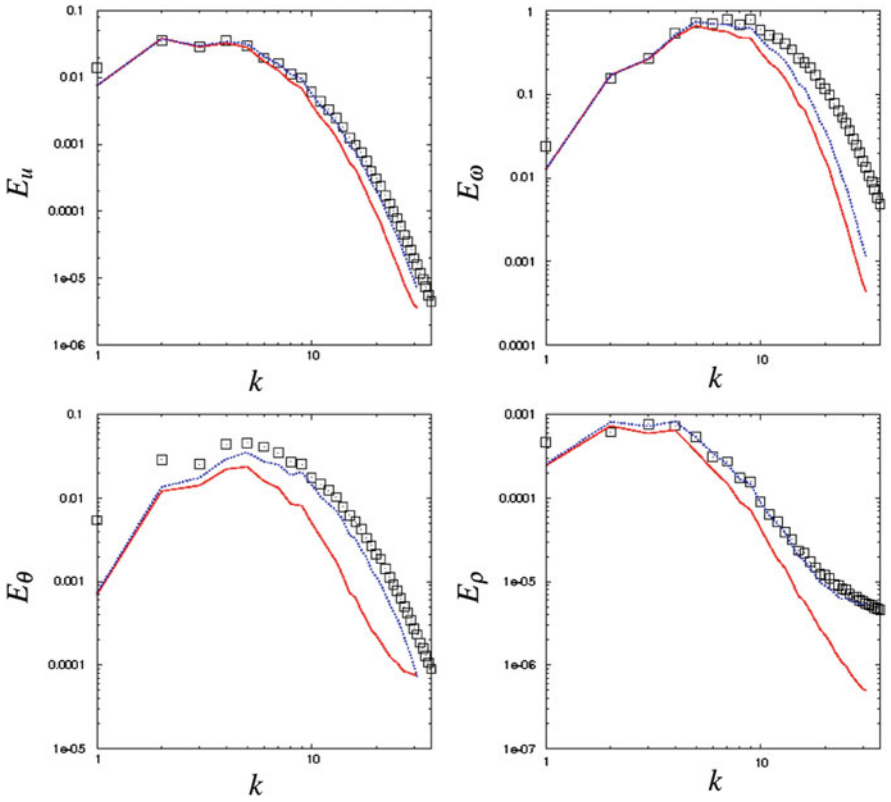


Fig. 2.11 Compressible isotropic turbulence. Spectra at $\frac{t}{\tau} = 4$ (logarithmic scales). The reference solution is the solution on a 256^3 grid (*squares*). WENO (*red*) and Hybrid (*blue*) solutions on 64^3 grid

References

1. Balsara, D.S., Shu, C.-W.: Monotonicity preserving weighted essentially non oscillatory schemes with increasingly high order of accuracy. J. Comput. Phys. **160**, 405–452 (2000)
2. Darian, H.M., Esfahanian, V., Hejranfar, K.: A shock-detecting sensor for filtering of high-order compact finite difference schemes. J. Comput. Phys. **230**, 494–514 (2011)
3. Ducros, F., Ferrand, V., Nicoud, F., Weber, C., Darracq, D., Gacherieu, C., Poinsot, T.: Large eddy simulation of the Shock-Turbulence interaction. J. Comput. Phys. **152**, 517–549 (1999)
4. Ducros, F., Laporte, F., Soulères, T., Guinot, V., Moinat, P., Caruelle, B.: High-order fluxes for conservative skew-symmetric-like schemes in structured meshes: application to compressible flows. J. Comput. Phys. **161**, 114–139 (2000)
5. Hill, D.J., Pullin, D.I.: Hybrid tuned center-difference-WENO method for larfe eddy simulations in the presence of strong shocks. J. Comput. Phys. **194**, 435–450 (2004)
6. Hill, D.J., Pantano, C., Pullin, D.I.: Larde-eddy simulation and multiscale modeling of a Richtmyer-Meshkov instability with reshock. J. Fluid Mech. **557**, 29–61 (2006)

7. Honein, A., Moin, P.: Higher entropy conservation and numerical stability of compressible turbulence simulations. J. Comput. Phys. **201**, 531–545 (2004)
8. Hu, X.Y., Wang, Q., Adams, N.A.: An adaptive central-upwind weighted essentially non-oscillatory scheme. J. Comput. Phys. **229**, 8952–8965 (2010)
9. Jiang, G.S., Shu, C.-W.: Efficient implementation of the weighted ENO schemes. J. Comput. Phys. **126**, 202–228 (1996)
10. Johnsen, E., Larsson, J., Bhagatwala, A.V., Cabot, W.H., Moin, P., Olson, B.J., Rawat, P.S., Shankar, S.K., Sjogreen, B., Yee, H.C., Zhong, X., Lele, S.K.: Assessment of high-resolution methods for numerical simulations of compressible turbulence with shock waves. J. Comput. Phys. **229**, 1213–1237 (2009)
11. Kennedy, C.A., Gruber, A.: Reduced aliasing formulations of the convective terms within the Navier–Stokes equations for a compressible fluid. J. Comput. Phys. **227**, 1676–1700 (2008)
12. Kok, J.C.: A high-order low-dispersion symmetry-presering finite-volume method for compressible flow on curvilinear grids. J. Comput. Phys. **18**, 6811–6832 (2009)
13. Kravchenko, A.G., Moin, P.: On the effect of numerical errors in large eddy simulations of turbulent flows. J. Comput. Phys. **131**, 310–322 (1997)
14. Lee, S., Lele, S.K., Moin, P.: Eddy shocklets in decaying compressible turbulence. Phys. Fluids A **3**, 657 (1991)
15. Li, G., Qiu, J.: Hybrid weighted essentially non-oscillatory schemes with different indicators. J. Comput. Phys. **229**, 8105–8129 (2010)
16. Martín, M.P., Taylor, E.M., Wu, M., Weirs, V.G.: A bandwidth-optimized WENO scheme for the effective direct numerical simulation of compressible turbulence. J. Comput. Phys. **220**, 270–289 (2006)
17. Morinishi, Y.: Skew-symmetric form of convective terms and fully conservative finite difference schemes for variable density low-mach number flows. J. Comput. Phys. **229**, 276–300 (2010)
18. Pirozzoli, S.: Generalized conservative approximations of split convective derivative operators. J. Comput. Phys. **229**, 7180–7190 (2010)
19. Shu, C.-W., Osher, S.: Efficient implementation of essentially non-oscillatory shock-capturing schemes. J. Comput. Phys. **77**, 439–471 (1988)
20. Tam, C.K.W., Webb, J.C.: Dispersion-relation-preserving finite difference schemes for computational acoustics. J. Comput. Phys. **107**, 262–281 (1993)
21. Taylor, E.M., Wu, M., Martín, M.P.: Optimization of nonlinear error for weighted essentially non-oscillatory methods in direct numerical simulations of compressible turbulence. J. Comput. Phys. **223**, 223 (2007)

Chapter 3
Finite Volume Formulation of a Third-Order Residual-Based Compact Scheme for Unsteady Flow Computations

Karim Grimich, Bertrand Michel, Paola Cinnella, and Alain Lerat

Abstract The paper discusses the design principles of a Finite-Volume Residual-Based Compact (RBC) scheme for the spatial discretization of the unsteady compressible governing equations of gas dynamics on general structured meshes. The scheme makes use of weighted approximations that allow to ensure high accuracy while taking benefit from the structured nature of the grid. The stability properties of the proposed spatial approximation are discussed. Numerical applications to unsteady compressible flows demonstrate the advantages of the proposed formulation with respect to straightforward extensions of RBC schemes.

3.1 Introduction

For the numerical simulation of fluid flows in complex geometries using body-fitted meshes, finite difference (FD) schemes require exact or approximate coordinate transformation techniques (see e.g. [12, 14]). The finite-volume (FV) method [11] represents an alternative to that. It is widely used in Computational Fluid Dynamics for its ability to handle complex geometries while ensuring conservation. FV schemes are classified according to the kind of mesh used (structured or unstructured) and to the place where the discrete unknown are assigned (cell centers, cell vertices, ...). The 3rd-order RBC scheme considered here was first developed for steady flows in FD formulation for Cartesian meshes and a direct (point-wise)

K. Grimich • P. Cinnella (✉) • A. Lerat
DynFluid Laboratory, Arts et Métiers-ParisTech, 151 bd. de l'Hopital, 75013 Paris, France
e-mail: paola.cinnella@ensam.eu; alain.lerat@ensam.eu

B. Michel
Département de Simulation Numérique des Ecoulements et Aéroacoustique, ONERA,
29 Avenue de la Division Leclerc, FR-92322 Chatillon Cedex, France
e-mail: bertrand.michel@onera.fr

R. Abgrall et al. (eds.), *High Order Nonlinear Numerical Schemes for Evolutionary PDEs*, Lecture Notes in Computational Science and Engineering 99,
DOI 10.1007/978-3-319-05455-1_3,

cell-centered FV formulation for curvilinear meshes [8, 9]. Then according to the general framework presented in [13], the accuracy of the FV formulation was improved for steady flow problems in [5]. Besides, it was extended to unstructured meshes in [3]. In the present paper, we continue the improvement of the accuracy of the FV formulation on structured meshes by dealing with the general case of unsteady flow problems. Straightforward extensions of the steady schemes to unsteady flows may lead to numerical instabilities for some flow conditions, as previously shown for FD RBC schemes [10]: precisely, even cross derivative terms appear in the truncation error of the numerical dissipation operator, leading to a negative dissipation coefficient over a range of wavenumbers; as a consequence, some frequencies of the numerical solutions may be amplified, leading eventually to a blow-up of the calculation. Such a behavior is highly configuration-dependent and hardly predictable. The finite volume extension of unsteady RBC schemes suffers from the same drawbacks. For this reason, in the following we design a dissipative FV RBC scheme for unsteady compressible flows of second order on highly deformed structured meshes and third order on slightly distorted meshes. We call this scheme RBCi where the i stands for *irregular*, since it is designed for irregular meshes. Because we are essentially interested into spatial accuracy properties, we restrict the discussion to semi-discrete schemes. RBCi is implemented within ONERAś flow solver elsA [1], and applied to the computation of unsteady flow problems.

The paper is organised as follows. In Sect. 3.2 we briefly recall some definitions for FV schemes. Section 3.3 gives the formulation of a Residual-Based Compact scheme in the FV framework. Section 3.4 describes two different FV extensions of the third-order accurate RBC scheme: the first one is a straightforward extension of the FD scheme, whereas the second one takes into account the effective mesh geometry in order to satisfy accuracy conditions in the FV sense on general meshes. In Sect. 3.6, we discuss the stability properties of this approximation when applied to regular Cartesian grids. Finally, Sect. 3.7 is devoted to numerical applications.

3.2 Finite Volume Framework

Consider the integral form of a hyperbolic system of conservation laws with initial conditions:

$$\frac{d}{dt}\int_{\Omega} w\, d\Omega + \oint_{\partial\Omega} \boldsymbol{\phi}.\mathbf{n}\, d\Gamma = 0, \quad w(x,y,z,0) = w_0(x,y,z) \tag{3.1}$$

where t is the time, x, y and z are Cartesian space coordinates, w is the vector of conservative variables, Ω is a closed control volume which boundary is $\partial\Omega$, $\boldsymbol{\phi}$ is the physical flux density $\boldsymbol{\phi} = [f, g, e]$ depending smoothly on w, and $\mathbf{n}$ is the

unit outward normal. The Jacobian matrices of the flux are denoted $A = \mathrm{d}f/\mathrm{d}w$, $B = \mathrm{d}g/\mathrm{d}w$ and $C = \mathrm{d}e/\mathrm{d}w$.

Define a structured mesh composed of hexaedral cells $\Omega_{j,k,l}$ with cell centers by $C_{j,k,l}$, cell vertices by $M_{j+\frac{1}{2},k+\frac{1}{2},l+\frac{1}{2}}$ and cell faces by $\Gamma_{j+\frac{1}{2},k,l}$, $\Gamma_{j,k+\frac{1}{2},l}$ or $\Gamma_{j,k,l+\frac{1}{2}}$:

$$\partial\Omega_{j,k,l} = \Gamma_{j+\frac{1}{2},k,l} \cup \Gamma_{j,k+\frac{1}{2},l} \cup \Gamma_{j,k,l+\frac{1}{2}} \cup \Gamma_{j-\frac{1}{2},k,l} \cup \Gamma_{j,k-\frac{1}{2},l} \cup \Gamma_{j,k,l-\frac{1}{2}}$$

The cell volume is denoted by $|\Omega_{j,k,l}|$ and an edge surface by $|\Gamma_{j+\frac{1}{2},k,l}|$. For each cell face $\Gamma_{j+\frac{1}{2},k,l}$ we denote $\mathbf{S}_{j+\frac{1}{2},k,l} = (\Gamma_x, \Gamma_y, \Gamma_z)_{j+\frac{1}{2},k,l}$ the oriented surface directed in the sense of increasing mesh indices, with module equal to $|\Gamma_{j+\frac{1}{2},k,l}|$. For each cell $\Omega_{j,k,l}$ we denote the cell center coordinates $(x_{j,k,l}, y_{j,k,l}, z_{j,k,l})$ and its maximum dimension in each direction $\delta x_{j,k,l}, \delta y_{j,k,l}$, and $\delta z_{j,k,l}$. Unknown vector w represents a pointwise value, located at cell centers. Finally, we define a characteristic mesh size by $h = \max(\max_{j,k,l} \delta x_{j,k,l}, \max_{j,k,l} \delta y_{j,k,l}, \max_{j,k,l} \delta z_{j,k,l})$.

Applied to the cell $\Omega_{j,k,l}$, Eq. (3.1) reads:

$$\frac{d}{dt}\int_{\Omega_{j,k,l}} w \, d\Omega + \sum_{\Gamma \in \partial\Omega_{j,k,l}} \int_{\Gamma} \boldsymbol{\phi}.\mathbf{n} \, d\Gamma = 0 \tag{3.2}$$

We define a FV exact residual R as:

$$R_{j,k,l} := \mathscr{V}_{j,k,l} + \mathscr{S}_{j,k,l} \tag{3.3}$$

$$\text{with} \qquad \mathscr{V}_{j,k,l} = \frac{d}{dt}\int_{\Omega_{j,k,l}} w \, d\Omega \quad \text{and} \quad \mathscr{S}_{j,k,l} = \sum_{\Gamma \in \partial\Omega_{j,k,l}} \int_{\Gamma} \boldsymbol{\phi}.\mathbf{n} \, d\Gamma. \tag{3.4}$$

A FV approximation of (3.3) can be written:

$$\tilde{R}_{j,k,l} = \tilde{\mathscr{V}}_{j,k,l} + \tilde{\mathscr{S}}_{j,k,l} \tag{3.5}$$

where $\tilde{\mathscr{V}}_{j,k,l}$ and $\tilde{\mathscr{S}}_{j,k,l}$ approximate the volume and the surface integrals respectively

$$\frac{1}{|\Omega_{j,k,l}|}\tilde{\mathscr{V}}_{j,k,l} = \frac{1}{|\Omega_{j,k,l}|}\mathscr{V}_{j,k,l} + \mathscr{O}(h^p), \quad \frac{1}{|\Gamma|}\tilde{\mathscr{S}}_{j,k,l} = \frac{1}{|\Gamma|}\mathscr{S}_{j,k,l} + \mathscr{O}(h^p) \tag{3.6}$$

for any edge Γ of the cell $\Omega_{j,k,l}$. If equations in (3.6) are satisfied simultaneously, the FV approximation (3.5) is said to be accurate at order p in the FV sense [13].

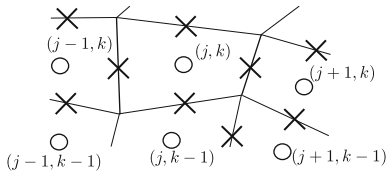


Fig. 3.1 Location of the discrete residuals, ∘: main residual $\tilde{R}$, ×: mid-point residual $\tilde{R}_1$ or $\tilde{R}_2$

3.3 Finite Volume RBC Schemes

Let us introduce the difference and average operators for any mesh function v:

$$(\delta_1 v)_{j+\frac{1}{2},k,l} = v_{j+1,k,l} - v_{j,k,l} \qquad (\mu_1 v)_{j+\frac{1}{2},k,l} = (v_{j+1,k,l} + v_{j,k,l})/2$$
$$(\delta_2 v)_{j,k+\frac{1}{2},l} = v_{j,k+1,l} - v_{j,k,l} \qquad (\mu_2 v)_{j,k+\frac{1}{2},l} = (v_{j,k+1,l} + v_{j,k,l})/2$$
$$(\delta_3 v)_{j,k,l+\frac{1}{2}} = v_{j,k,l+1} - v_{j,k,l} \qquad (\mu_3 v)_{j,k,l+\frac{1}{2}} = (v_{j,k,l+1} + v_{j,k,l})/2$$

where j, k and l are integers or half integers. A *residual-based* scheme can be expressed in terms of approximations of the exact residual R. More precisely, such a scheme is of the following form:

$$(\tilde{R}_0)_{j,k,l} = \tilde{d}_{j,k,l} \tag{3.7}$$

where $\tilde{R}_0$ is a space-centered approximation of R, called the *main residual* and $\tilde{d}$ is a residual-based dissipation term defined in terms of first-order differences of the residual as:

$$\tilde{d}_{j,k,l} = \frac{1}{2}[\delta_1(\Phi_1 \tilde{R}_1) + \delta_2(\Phi_2 \tilde{R}_2) + \delta_3(\Phi_3 \tilde{R}_3)]_{j,k,l} \tag{3.8}$$

where $\tilde{R}_1$, $\tilde{R}_2$ and $\tilde{R}_3$, respectively defined at $j+\frac{1}{2},k,l$, $j,k+\frac{1}{2},l$ and $j,k,l+\frac{1}{2}$ (cf. Fig. 3.1), are approximations of R on shifted control volumes, are called the *mid-point residuals*, and Φ_1, Φ_2, Φ_3 are numerical viscosity matrices (defined at the same location as the corresponding mid-point residuals). These matrices depend only on the eigensystems of the Jacobian matrices A, B and C and on the geometry of the current cell $\Omega_{j,k,l}$. They are designed once for all [10] and use no tuning parameters nor limiters. Since these matrices remain $\mathcal{O}(1)$ as h tend to zero, the dissipation $\tilde{d}$ represents a numerical approximation of the partial differential term:

$$d = \left[\frac{\delta x}{2}(\Phi_1 R)_x + \frac{\delta y}{2}(\Phi_2 R)_y + \frac{\delta z}{2}(\Phi_3 R)_z\right] \tag{3.9}$$

This leading term of the expansion, that is only first order, vanishes for an exact solution ($R = 0$), so that $\tilde{d}$ is actually consistent with a higher-order dissipation term. For FD RBC schemes, the accuracy and spectral properties of the RBC

dissipation operator have been discussed in [10] and [4], respectively. Its properties in the case of FV extensions of RBC schemes are discussed in this paper.

3.4 Direct FV Extension of RBC Schemes

As a first step, we describe a straightforward FV extension of the FD RBC schemes of nominal 3rd, 5th or 7th order of accuracy for unsteady flows, referred-to as RBCq ($q = 3, 5, 7$) in the following. These *non-weighted* versions of the schemes degenerate on the corresponding FD schemes [10], when applied to regular Cartesian grids.

3.4.1 Approximation of the Main Residual

The main residual approximation $\tilde{R}_0$ can be splitted into a volume integral and a surface integral approximation

$$(\tilde{R}_0)_{j,k,l} = (\tilde{\mathscr{V}_0})_{j,k,l} + (\tilde{\mathscr{S}_0})_{j,k,l} \tag{3.10}$$

where $\tilde{\mathscr{V}_0}$ is a centered approximation of $\mathscr{V}$ and $\tilde{\mathscr{S}_0}$ is a centered approximation of $\mathscr{S}$, $\mathscr{V}$ and $\mathscr{S}$ being defined by Eq. (3.4). Precisely, $\tilde{\mathscr{V}_0}$ and $\tilde{\mathscr{S}_0}$ are given by the following compact approximation operators:

$$(\tilde{\mathscr{V}_0})_{j,k,l} = \left[\mathscr{FD}^{w_0}_{RBCq}(w_t)\right]_{j,k,l} |\Omega_{j,k,l}| \text{ and } (\tilde{\mathscr{S}_0})_{j,k,l} = [\delta_1 F_1 + \delta_2 F_2 + \delta_3 F_3]_{j,k,l} \tag{3.11}$$

with, e.g.:
$$(F_1)_{j+\frac{1}{2},k,l} = \left[\mathscr{FD}^{F_{01}}_{RBCq}(\boldsymbol{\phi})\right]_{j+\frac{1}{2},k,l} \cdot \mathbf{S}_{j+\frac{1}{2},k,l}, \tag{3.12}$$

where $\mathscr{FD}^{w_0}_{RBCq}$ and $\mathscr{FD}^{F_{01}}_{RBCq}$ are FD operators applied to the time derivative and to the flux in the first direction, given in [10] for an RBCq scheme. For instance, for RBC3:

$$\mathscr{FD}^{w_0}_{RBC3}[.] = \left(I + \frac{1}{6}\delta_1^2\right)\left(I + \frac{1}{6}\delta_2^2\right)\left(I + \frac{1}{6}\delta_3^2\right)[.] \tag{3.13}$$

$$\mathscr{FD}^{F_{01}}_{RBC3}[.] = \left(I + \frac{1}{6}\delta_2^2\right)\left(I + \frac{1}{6}\delta_3^2\right)\mu_1[.] \tag{3.14}$$

Similar expressions are used for $(F_2)_{j,k+\frac{1}{2},l}$ and $(F_3)_{j,k,l+\frac{1}{2}}$.

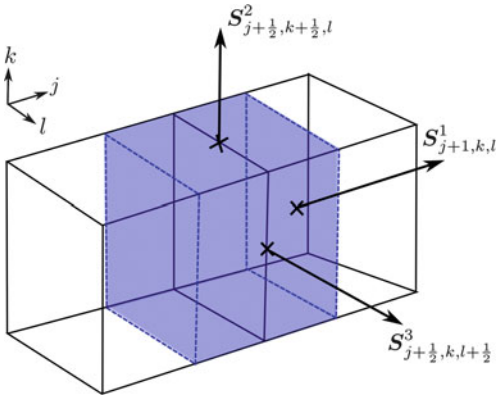


Fig. 3.2 Shifted cell used for the direct FV extension

3.4.2 Residual-Based Dissipation Operator

To complete the formulation of the dissipation operator $\tilde{d}_{j,k,l}$, defined by Eq. (3.8). This requires approximations of the mid-point residuals. To this end, we define shifted control volumes around cell faces; for instance, a shifted volume denoted $\Omega_{j+\frac{1}{2},k,l}$ will be used to approximate $(\tilde{R}_1)_{j+\frac{1}{2},k,l}$. We associate to this shifted cell the oriented surfaces in the directions 1, 2 and 3 (cf. Fig. 3.2):

$$\mathbf{S}^1_{j,k,l} = \frac{1}{2}(\mathbf{S}_{j+\frac{1}{2},k,l} + \mathbf{S}_{j-\frac{1}{2},k,l}) \quad \mathbf{S}^2_{j+\frac{1}{2},k+\frac{1}{2},l} = \frac{1}{2}(\mathbf{S}_{j,k+\frac{1}{2},l} + \mathbf{S}_{j+1,k+\frac{1}{2},l}) \tag{3.15}$$

$$\mathbf{S}^3_{j+\frac{1}{2},k,l+\frac{1}{2}} = \frac{1}{2}(\mathbf{S}_{j,k,l+\frac{1}{2}} + \mathbf{S}_{j+1,k,l+\frac{1}{2}})$$

Mid-point residuals can also be splitted into a mid-point volume integral and a mid-point surface integral. For instance, the mid-point residual in the direction 1 writes:

$$(\tilde{R}_1)_{j+\frac{1}{2},k,l} = (\tilde{\mathscr{V}}_1)_{j+\frac{1}{2},k,l} + (\tilde{\mathscr{S}}_1)_{j+\frac{1}{2},k,l} \tag{3.16}$$

where $\tilde{\mathscr{V}}_1$ and $\tilde{\mathscr{S}}_1$ are given by the following approximation operators:

$$\begin{aligned}(\tilde{\mathscr{V}}_1)_{j+\frac{1}{2},k,l} &= \left[\mathscr{F}\mathscr{D}^{w_1}_{RBCq}(w_l)\right]_{j+\frac{1}{2},k,l} |\Omega_{j+\frac{1}{2},k,l}| \\ (\tilde{\mathscr{S}}_1)_{j+\frac{1}{2},k,l} &= \left[\mathscr{F}\mathscr{D}^{F_{11}}_{RBCq}(\boldsymbol{\phi}).\mathbf{S}^1 + \mathscr{F}\mathscr{D}^{F_{12}}_{RBCq}(\boldsymbol{\phi}).\mathbf{S}^2 + \mathscr{F}\mathscr{D}^{F_{13}}_{RBCq}(\boldsymbol{\phi}).\mathbf{S}^3\right]_{j+\frac{1}{2},k,l}\end{aligned} \tag{3.17}$$

where $|\Omega|_{j+\frac{1}{2},k,l} = \frac{1}{2}(|\Omega|_{j,k,l} + |\Omega|_{j+1,k,l})$ and $\mathscr{F}\mathscr{D}^{w_1}_{RBCq}$, $\mathscr{F}\mathscr{D}^{F_{11}}_{RBCq}$, $\mathscr{F}\mathscr{D}^{F_{12}}_{RBCq}$ and $\mathscr{F}\mathscr{D}^{F_{13}}_{RBCq}$ are the FD operators used in the FD definition of RBCq scheme [10]. For RBC3, these write:

$$\mathscr{F}\mathscr{D}^{w_1}_{RBC3}[.] = \left(I + \frac{1}{6}\delta_2^2\right)\left(I + \frac{1}{6}\delta_3^2\right)\mu_1[.] \quad \mathscr{F}\mathscr{D}^{F_{13}}_{RBC3}[.]$$
$$= \left(I + \frac{1}{6}\delta_2^2\right)\delta_3\mu_3\mu_1[.]$$
$$\mathscr{F}\mathscr{D}^{F_{11}}_{RBC3}[.] = \left(I + \frac{1}{6}\delta_2^2\right)\left(I + \frac{1}{6}\delta_3^2\right)\delta_1[.] \quad \mathscr{F}\mathscr{D}^{F_{12}}_{RBC3}[.]$$
$$= \left(I + \frac{1}{6}\delta_3^2\right)\delta_2\mu_2\mu_1[.]$$

$\tilde{R}_2$ and $\tilde{R}_3$ use similar approximations in the other directions of the mesh corresponding to the FD operator introduced in [10]. Similar formulations can be easily derived for RBC5 and RBC7. Reference [13] shows, in the framework of non compact schemes, that nominal accuracy can still be preserved if the mesh satisfies suitable regularity conditions. The higher the scheme accuracy, the more stringent grid regularity requirements, so that nominal accuracy is generally lost for cases of practical interest. In the following, this extension is referred-to as RBCr for RBC scheme of nominal 3rd order of accuracy (r stands for regular). In the next Section, we introduce a FV extension of RBC3, called RBCi, which is at least second-order accurate on any structured mesh and third-order on meshes with smooth irregularities.

3.5 Finite Volume Third-Order RBC Scheme

RBCi is also expressed through the formula (3.7) with a numerical dissipation of the form (3.8) and the residual is defined in (3.3). The difference with respect to RBCr scheme described above is that the main and mid-point residuals are now approximated by means of weighted discretization operators that take into account mesh deformations. The procedure used to approximate the volume and surface integrals involved in the residuals is described hereafter. The dissipation matrices Φ_i do not affect the order of accuracy of the scheme and they are consequently not modified.

3.5.1 Approximation of the Main Residual

Once again, we express the main residual as the sum of a volume integral and a surface integral. For the sake of simplicity and to avoid complex reconstruction procedures we do not try to ensure 3rd order accuracy on any irregular mesh. Instead, we look for approximations that are 2nd order accurate on highly irregular grids and third order accurate on sufficiently smooth grids. Grid regularity criteria

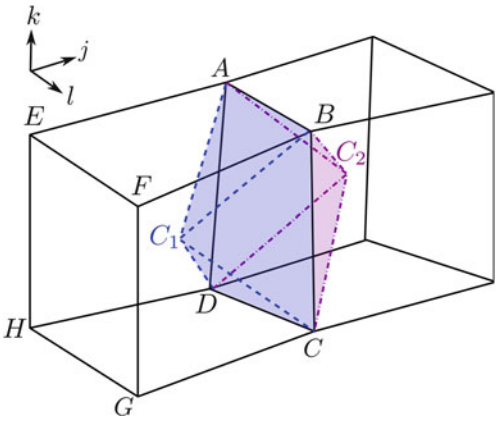


Fig. 3.3 Octahedral shifted cell $\Omega_{j+1/2,k,l}$

were introduced in [13] based on grid stretching, parallelism and orthonormality criteria.

Let us start with the approximation of the volume integral in the main residual $\mathscr{V}_0$. The standard second-order accurate approximation for $\mathscr{V}_0$ at point (j,k,l) is:

$$(\tilde{\mathscr{V}_0})^{std} = w_{t\,P}\,|\Omega_{j,k,l}| = (\mathscr{V}_0)_{j,k,l} + \mathscr{O}(|\Omega_{j,k,l}|h^2), \tag{3.18}$$

where P is any point in $\Omega_{j,k,l}$. Hereafter, we choose instead to approximate $(\mathscr{V}_0)_{j,k,l}$ as a linear combination of the standard volume integral referred to the cell-center and of integrals evaluated at cell vertices A, B, C, D, E, F, G and H (see Fig. 3.3):

$$\begin{aligned}\tilde{\mathscr{V}_0} =& \alpha T_{C_1} + \beta(T_A + T_B + T_C + T_D + T_E + T_F + T_G + T_H)\\ =& \mathscr{V}_0 + \mathscr{O}(|\Omega_{j,k,l}|h^2)\end{aligned} \tag{3.19}$$

where we set $T_P = w_{t\,P}\,|\Omega_{j,k,l}|$, and the values of w_t at the cell vertices are obtained by 2nd-order linear interpolation from the surrounding cell centers. Since T_P is a 2nd-order accurate approximation of $\mathscr{V}_0$, Eq. (3.19) is also 2nd-order accurate for general irregular grids. An additional relation is required to complete the choice of α and β. To this purpose, we develop Eq. (3.19) in the case of a uniform Cartesian grid. In this case, the interpolated values of w_t at cell vertices are just given by $1/8$ of the sum of the surrounding cell-center values. This expression is written under the form:

$$\frac{1}{|\Omega_{j,k,l}|}\tilde{\mathscr{V}_0}^{Cart} = \left\{\left[\mathscr{F}\mathscr{D}^{w_0}_{RBC3} + \gamma_1(\delta_1^2\delta_2^2 + \delta_1^2\delta_3^2 + \delta_2^2\delta_3^2) + \gamma_2\delta_1^2\delta_2^2\delta_3^2\right](w_t)\right\}_{j,k,l} \tag{3.20}$$

where γ_1 and γ_2 are free coefficients introduced for convenience. Equation (3.20) is the non-weighted approximation of the main volume integral (3.11), supplemented

with some higher order terms resulting from the general formulation (3.19). By identification between Eqs. (3.19) and (3.20), we get the unique solution

$$\alpha = \frac{1}{3}, \quad \beta = \frac{1}{12}, \quad \gamma_1 = \frac{1}{72} \quad \text{and} \quad \gamma_2 = \frac{5}{864}. \tag{3.21}$$

In practice, the approximation for the volume integral on a general grid used in the RBCi formulation is given by relation (3.19), with $\alpha = 1/3$ and $\beta = 1/12$. We now look for an approximation of the surface integral $(\mathscr{S}_0)_{j,k,l}$. This requires the evaluation of flux integrals over each cell face. To ensure accuracy on highly distorted grids, one important point is to account for possible skewness of the faces. To this purpose, we consider the auxiliary shifted control volume represented in Fig. 3.3. For any main cell face identified through its vertices A, B, C and D, this control volume is represented by an irregular octahedron with vertices coincident with the face vertices plus the adjacent cell centers C_1 and C_2. To fix ideas, we consider the approximation of the surface integral over face $\Gamma_{j+\frac{1}{2},k,l}$, so that the vertices A, B, C and D correspond to points $(j+\frac{1}{2}, k+\frac{1}{2}, l+\frac{1}{2})$, $(j+\frac{1}{2}, k-\frac{1}{2}, l+\frac{1}{2})$, $(j+\frac{1}{2}, k-\frac{1}{2}, l-\frac{1}{2})$ and $(j+\frac{1}{2}, k+\frac{1}{2}, l-\frac{1}{2})$. Integrals on the other faces are approximated in a similar way. The surface integral over face $\Gamma_{j+\frac{1}{2},k,l}$ is approximated as follows:

$$\int_{\Gamma_{j+\frac{1}{2},k,l}} \boldsymbol{\phi} \cdot \mathbf{n} d\Gamma = \frac{1}{2}(I_L - I_R) \tag{3.22}$$

where $$I_L = \int_{\Gamma_{j+\frac{1}{2},k,l}} \boldsymbol{\phi} \cdot \mathbf{n}_{P_1} d\Gamma, \; I_R = \int_{\Gamma_{j+\frac{1}{2},k,l}} \boldsymbol{\phi} \cdot \mathbf{n}_{P_2} d\Gamma$$

and $\mathbf{n}_{P_1}$ and $\mathbf{n}_{P_2}$ are the outward normals to the pyramids $ABCDC_1$ and $ABCDC_2$, respectively. On face $\Gamma_{j+\frac{1}{2},k,l} = ABCD$ we have $\mathbf{n}_{P_1} = \mathbf{n}$ and $\mathbf{n}_{P_2} = -\mathbf{n}$. Instead of calculating these integrals on face $\Gamma_{j+\frac{1}{2},k,l}$, which can be non planar for a general irregular mesh, we express them as a combination of integrals on the other faces of the pyramids, which are triangular and then belong to a plane. To this purpose, we remark that, for steady flows, application of the conservation laws to pyramids $ABCDC_1$ and $ABCDC_2$, respectively, leads to:

$$I_L = -\int_{AC_1B} \boldsymbol{\phi} \cdot \mathbf{n}_{P_1} d\Gamma - \int_{BC_1C} \boldsymbol{\phi} \cdot \mathbf{n}_{P_1} d\Gamma - \int_{CC_1D} \boldsymbol{\phi} \cdot \mathbf{n}_{P_1} d\Gamma - \int_{DC_1A} \boldsymbol{\phi} \cdot \mathbf{n}_{P_1} d\Gamma \tag{3.23}$$

$$I_R = -\int_{BC_2A} \boldsymbol{\phi} \cdot \mathbf{n}_{P_2} d\Gamma - \int_{CC_2B} \boldsymbol{\phi} \cdot \mathbf{n}_{P_2} d\Gamma - \int_{DC_2C} \boldsymbol{\phi} \cdot \mathbf{n}_{P_2} d\Gamma - \int_{AC_2D} \boldsymbol{\phi} \cdot \mathbf{n}_{P_2} d\Gamma \tag{3.24}$$

Relations (3.23) and (3.24) are exact for steady flows and I_L is strictly equal to $-I_R$ for an exact solution. For unsteady flows, it can be shown that:

$$\frac{1}{2}(I_L - I_R) = \int_{\Gamma_{j+\frac{1}{2},k,l}} \boldsymbol{\phi} \cdot \mathbf{n} d\Gamma + |\Gamma_{j+\frac{1}{2},k,l}| \mathscr{O}(h^2) \tag{3.25}$$

This demonstrates that (3.22)–(3.24) can be used to estimate surface integrals in the main residual exactly for steady flows and to the 2nd order on highly irregular grids. We still need to approximate the surface integrals on triangles used in relations (3.23) and (3.24). To this purpose, we first interpolate linearly the fluxes at cell vertices A, B, C and D, and then approximate the integrals as:

$$\int_{AC_1B} \boldsymbol{\phi} \cdot \mathbf{n}_{P_1} d\Gamma = \frac{1}{3}(\boldsymbol{\phi}_A + \boldsymbol{\phi}_{C_1} + \boldsymbol{\phi}_B) \cdot \mathbf{n}_{AC_1B}, \tag{3.26}$$

which is 2nd-order accurate. Similar approximations as (3.26) are used for the other triangles. The advantage of the proposed approach is that triangular surfaces allow to compute a unique normal vector to that surface, which greatly enhances the scheme accuracy on highly skewed grids. Finally, thanks to the choice of a weighting coefficients of $1/3$ in (3.26), on a Cartesian grid $\tilde{\mathscr{S}}_0$ degenerates to Eq. (3.11) with

$$(F_1)_{j+\frac{1}{2},k,l} = \left[\mathscr{F}\mathscr{D}^{F_{01}}_{RBC3}(\boldsymbol{\phi}) + \frac{1}{72}\delta_2^2\delta_3^2\mu_1\boldsymbol{\phi} \right]_{j+\frac{1}{2},k,l} \cdot \mathbf{S}_{j+\frac{1}{2},k,l}. \tag{3.27}$$

This is the non-weighted of RBC3 operator augmented with a high-order corrective term. In summary the final approximation of the main residual used for RBCi is:

$$\tilde{R}_0 = \tilde{\mathscr{V}}_0 + \tilde{\mathscr{S}}_0 \tag{3.28}$$

with $\tilde{\mathscr{V}}_0$ given by (3.19) with coefficients (3.21) and $\tilde{\mathscr{S}}_0$ given by a sum of integrals of the form (3.22). Since the approximations used for $\tilde{\mathscr{V}}_0$ and $\tilde{\mathscr{S}}_0$ are second-order accurate on general grids, then $\tilde{R}_0$ is at least second-order accurate in a FV sense.

3.5.2 Approximation of the Mid-point Residuals

To fix ideas, we consider the approximation of mid-point residual R_1 at point $(j + \frac{1}{2}, k, l)$, the other residuals being approximated in a similar way. To evaluate R_1 at $(j + \frac{1}{2}, k, l)$, we consider again the shifted octahedral control volume of Fig. 3.3, and look for an approximation $\tilde{R}_1$ of the form:

$$(\tilde{R}_1)_{j+\frac{1}{2},k,l} = \left(\tilde{\mathscr{V}}_1 + \tilde{\mathscr{S}}_1\right)_{j+\frac{1}{2},k,l} \tag{3.29}$$

with $\tilde{\mathscr{V}}_1$ an approximation of

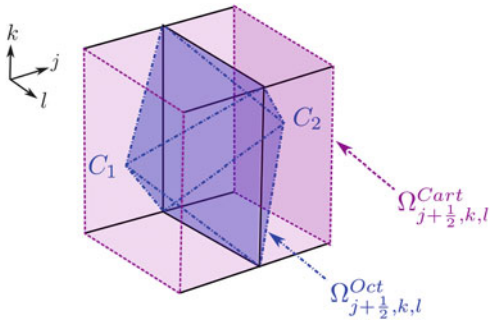


Fig. 3.4 Volumes $\Omega^{Oct}_{j+\frac{1}{2},k,l}$ and $\Omega^{Cart}_{j+\frac{1}{2},k,l}$

$$\mathscr{V}_1 = 3\int_{\Omega^{Oct}_{j+\frac{1}{2},k,l}} w_t \, d\Omega = 3\left(\int_{C_1ABCD} w_t \, d\Omega + \int_{C_2ABCD} w_t \, d\Omega\right), \tag{3.30}$$

where we denote $\Omega^{Oct}_{j+\frac{1}{2},k,l}$ the shifted control volume and we split the volume integral on the shifted cell in two integrals over the pyramids on both sides. We denote $\tilde{\mathscr{S}_1}$ an approximation of:

$$\mathscr{S}_1 = 3\left(\oint_{ABCDC_1} \boldsymbol{\phi} \cdot \mathbf{n}_{P_1} d\Gamma + \oint_{ABCDC_2} \boldsymbol{\phi} \cdot \mathbf{n}_{P_2} d\Gamma\right) \tag{3.31}$$

The factor 3 in (3.30) and (3.31) accounts for the fact that the volume of an octahedron is one third that of a parallelepiped with the same height C_1C_2 (see Fig. 3.4), i.e. the shifted cell used for the FV RBCr scheme. To find $\mathscr{V}_1$, we look for an approximation of the volume integrals on $ABCDC_1$ and $ABCDC_2$ under the form:

$$\begin{aligned}
\int_{C_1ABCD} w_t \, d\Omega &= |\Omega_{P_1}|[\alpha^\star w_{t\,C_1} + \beta^\star(w_{t\,A} + w_{t\,B} + w_{t\,C} + w_{t\,D}) + \mathscr{O}(h^2)] \\
\int_{C_2ABCD} w_t \, d\Omega &= |\Omega_{P_2}|[\alpha^\star w_{t\,C_2} + \beta^\star(w_{t\,A} + w_{t\,B} + w_{t\,C} + w_{t\,D}) + \mathscr{O}(h^2)]
\end{aligned} \tag{3.32}$$

where $\alpha^\star$ and $\beta^\star$ are real coefficients and $|\Omega_{P_1}|$ and $|\Omega_{P_2}|$ are the volumes of pyramids $ABCDC_1$ and $ABCDC_2$, respectively. By plugging (3.32) in (3.30) we obtain:

$$\tilde{\mathscr{V}_1} = \mathscr{V}_1 + \mathscr{O}(|\Omega_{j+\frac{1}{2},k,l}|h^2) \tag{3.33}$$

with
$$\begin{aligned}
\tilde{\mathscr{V}_1} =& 3\alpha^\star(|\Omega_{P_1}|w_{t\,C_1} + |\Omega_{P_2}|w_{t\,C_2}) \\
&+ 3\beta^\star(|\Omega_{P_1}| + |\Omega_{P_2}|)(w_{t\,A} + w_{t\,B} + w_{t\,C} + w_{t\,D})
\end{aligned} \tag{3.34}$$

We still need to choose the weighting coefficients $\alpha^\star$ and $\beta^\star$. Consistency requires that $\alpha^\star + 4\beta^\star = 1$. To this end we follow a procedure similar to that used to construct $\tilde{\mathscr{V}}_0$, and we look for a set of coefficients such that $\tilde{\mathscr{V}}_1$ can be reduced to the following writing in the case of a regular Cartesian grid:

$$\tilde{\mathscr{V}}_1 = \left\{\left[\mathscr{F}\mathscr{D}^{w_1}_{RBC3} + \gamma^\star \delta_2^2 \delta_3^2 \mu_1\right](w_t)\right\}_{j+\frac{1}{2},k,l} |\Omega^{Cart}_{j+\frac{1}{2},k,l}| \tag{3.35}$$

where $\gamma^\star$ is a supplementary coefficient introduced for convenience and $|\Omega^{Cart}_{j+\frac{1}{2},k,l}|$ is the volume of the Cartesian shifted cell of Figs. 3.2 and 3.4. Similarly to the approximation of $\tilde{\mathscr{V}}_0$, relation (3.35) is the sum of a term formally identical to the non weighted approximation RBCr, plus an additional higher-order term with coefficient $\gamma^\star$, required to match the general second-order formulation of the scheme given by (3.34). By developing (3.34) and (3.35) in the case of a uniform Cartesian grid, by identification of analogous terms we get the unique choice of coefficients

$$\alpha^\star = \frac{1}{3}, \quad \beta^\star = \frac{1}{6} \quad \text{and} \quad \gamma^\star = \frac{1}{72}. \tag{3.36}$$

The approximation of $\mathscr{V}_1$ is then given by Eq. (3.34) with the coefficients (3.36).

Finally, we turn to the approximation of $\mathscr{S}_1$. Note that:

$$\mathscr{S}_1 = 3\left(\oint_{ABCDC_1} \boldsymbol{\phi}\cdot\mathbf{n}_{P_1} d\Gamma + \oint_{ABCDC_2} \boldsymbol{\phi}\cdot\mathbf{n}_{P_2} d\Gamma\right) = 3(I_L + I_R) \tag{3.37}$$

where I_L and I_R are given by Eqs. (3.23) and (3.24) and are computed by following the procedure described in Sect. 3.5.1, which leads to a second order-accurate approximation. For a regular Cartesian grid, Eq. (3.37) reduces to the RBCr formulation, augmented with higher-order terms specific to the second-order formulation on irregular grids. The mid-point residual $\tilde{R}_1$ writes as (3.29) with (3.37) and (3.34) defined by the coefficients (3.36). Similar expressions are used in the second and third direction to construct $\tilde{R}_2$ and $\tilde{R}_3$ with the same choice of coefficients $\alpha^\star$ and $\beta^\star$.

3.6 Stability Properties of RBC*i*

Stability properties of FV RBC schemes depend on mesh regularity properties and may hardly be studied in the general case. For regular Cartesian meshes, stability and spectral properties of FV schemes may be studied using analysis tools typical of the FD framework. Since RBCr degenerates to its FD equivalent when written on Cartesian grids, results discussed in [10] for FD RBC3 remain also valid for RBCr.

3.6.1 Dissipation Properties of RBC*i*

Similarly to all RBC schemes, RBC*i* is the sum of a purely centered, non dissipative, approximation of the residual, $\tilde{R}_0$, and a dissipation term $\tilde{d}$. Since $\tilde{R}_0$ does not introduce any form of dissipation, we investigate the dissipation properties of RBC*i* by studying the discrete operator $\tilde{d}_{j,k,l}$. According to the study carried out for FD RBC schemes of order $2p-1$ [10], the operator (3.8) with the mid-point residuals of RBC*r* is always dissipative. We prove hereafter that the same is true when the residuals of RBC*i* are used, since this introduces an additional dissipation with respect to RBC*r*. To this purpose, we apply RBC*r* and RBC*i* to the linear scalar problem:

$$w_t + Aw_x + Bw_y + Cw_z = 0 \qquad A, B, C \in \mathbb{R} \tag{3.38}$$

In this case the dissipation matrices Φ_1, Φ_2, Φ_3 satisfy the conditions

$$\Phi_1 A \geqslant 0, \quad \Phi_2 B \geqslant 0, \quad \Phi_3 C \geqslant 0, \quad \delta x \Phi_1 B = \delta y \Phi_2 A, \quad \delta x \Phi_1 C = \delta z \Phi_3 A \tag{3.39}$$

with
$$\Phi_1 = \mathrm{sgn}(A)\phi, \quad \Phi_2 = \mathrm{sgn}(B)\psi, \quad \Phi_3 = \mathrm{sgn}(C)\zeta,$$
$$\phi = \min\left(1, \frac{1}{\alpha}, \frac{1}{\beta}\right), \psi = \min\left(1, \alpha, \frac{\alpha}{\beta}\right), \zeta = \min\left(1, \beta, \frac{\beta}{\alpha}\right) \tag{3.40}$$

with
$$\alpha = \frac{\delta x |B|}{\delta y |A|} \text{ and } \beta = \frac{\delta x |C|}{\delta z |A|}.$$

The interested reader is referred to [8, 10] for more details about the dissipation matrices Φ_1, Φ_2, Φ_3 and their extension to systems of conservation laws.

On the other hand, the Taylor-series expansion of $\tilde{d}^r$ (see [10] for details) writes as:

$$\frac{1}{|\Omega|}(\tilde{d}^r) = D^r + \mathscr{O}(h^4) \tag{3.41}$$

with
$$D^r = -\frac{1}{24}\left[\delta x(\Phi_1 \delta x^2 f_{xxx})x + \delta y(\Phi_2 \delta y^2 g_{yyy})_y + \delta z(\Phi_3 \delta z^2 e_{zzz})_z\right] \tag{3.42}$$

For problem (3.38), Φ_1, Φ_2 and Φ_3 are scalar constants, and (3.41) reduces to

$$D^r = \mathscr{D}^r w$$

with $\mathscr{D}^r$ a 4th-order linear partial differential operator. The Fourier symbol of $\mathscr{D}^r$ is:

$$\hat{\mathscr{D}}^r = -\frac{1}{24}\left[\delta x(\Phi_1 A \delta x^2 \xi^4) + \delta y(\Phi_2 B \delta y^2 \eta^4) + \delta z(\Phi_3 C \delta z^2 \zeta^4)\right] \tag{3.43}$$

where ξ, η and ζ are the reduced wave numbers in the x, y, and z direction, respectively. All the derivatives in $\mathscr{D}^r$ being even, $\hat{\mathscr{D}}^r(\xi, \eta, \zeta)$ is real. The operator D^r is dissipative, in the sense that [10]:

$$\forall \xi \in \mathbb{R}, \forall \eta \in \mathbb{R}, \forall \zeta \in \mathbb{R}, \qquad \hat{\mathscr{D}}^r(\xi, \eta, \zeta) \leqslant 0. \tag{3.44}$$

In other terms, D^r satisfies the following

Dissipation property. *The operator* (3.42) *is dissipative for any advection direction (A,B,C) and any functions* Φ_1, Φ_2 *and* Φ_3 *satisfying the conditions* (3.39).

We now turn to RBCi. In this case, the dissipation operator $\tilde{d}^i$ can be written in a way similar to Eq. (3.41). We consider its Fourier symbol:

$$\hat{\mathscr{D}}^i = \hat{\mathscr{D}}^r - \frac{\delta x^3}{12}\Phi_1 A \xi^4 \mathscr{D}' \tag{3.45}$$

where

$$\mathscr{D}' = \sigma^2 + \omega^2 + \varphi^2\sigma^2 + \varphi^2\sigma^2\omega^2 + \psi^2\omega^2 + \psi^2\sigma^2\omega^2 - 2\left(\varphi\sigma\omega^2 + \psi\sigma^2\omega + \varphi\psi\sigma\omega\right)$$

with $\sigma = \dfrac{\delta y}{\delta x}\dfrac{\eta}{\xi}$, $\omega = \dfrac{\delta z}{\delta x}\dfrac{\zeta}{\xi}$, $\varphi = \dfrac{\Phi_2}{\Phi_1} = \dfrac{\delta x B}{\delta y A}$ and $\psi = \dfrac{\Phi_3}{\Phi_1} = \dfrac{\delta x C}{\delta z A}$. Since:

$$\forall \sigma, \psi, \omega \text{ and } \varphi \in \mathbb{R}, \qquad \mathscr{D}' = \sigma^2(1-\psi\omega)^2 + \omega^2(1-\varphi\sigma)^2 + (\varphi\sigma - \psi\omega)^2 \geq 0$$

it appears that corrective terms in RBCi introduce additional dissipation with respect to RBCr, so that the preceding dissipation property is a fortiori valid for RBCi.

3.6.2 Cauchy Stability of RBCi

We now prove that RBCi is a Cauchy-stable semi discrete scheme. Hereafter, we recall the Cauchy stability condition in the case of a linear scalar problem of the form (3.38). The Fourier transform of (3.38) is

$$\hat{w}_t = -i(Ak_x + Bk_y + Ck_z)\hat{w} = -i\mathbf{A}\cdot\mathbf{k}\hat{w} \tag{3.46}$$

where $\mathbf{A} = (A, B, C)$ is the advection velocity vector and $\mathbf{k} = (k_x, k_y, k_z)$ a 3D wave vector. Denoting $\dot{\mathbf{A}} = (\dot{A}, \dot{B}, \dot{C})$ a vector whose components are the CFL numbers in the x, y and z directions, respectively:

$$\dot{A} = A\Delta t/\delta x, \quad \dot{B} = B\Delta t/\delta y, \quad \dot{C} = C\Delta t/\delta z$$

and introducing the reduced wave number in the $\dot{\mathbf{A}}$-direction:

$$\xi_{\theta,\phi} = \frac{\dot{\mathbf{A}}}{|\dot{\mathbf{A}}|} \cdot \boldsymbol{\xi}$$

with $\boldsymbol{\xi} = (\xi, \eta, \zeta) = (k_x\delta x, k_y\delta y, k_z\delta z)$ the reduced wave vector, Eq. (3.46) can be rewritten as:

$$\hat{w}_t = -i\,|\dot{\mathbf{A}}|\xi_{\theta,\phi}\frac{\hat{w}}{\Delta t} \tag{3.47}$$

The notation $\xi_{\theta,\phi}$ refers to the local advection direction with respect to the mesh, where $\cos(\phi) = \dot{C}/|\dot{\mathbf{A}}|$, $\tan(\theta) = \dot{B}/\dot{A}$, and $|\dot{\mathbf{A}}|$ is the global CFL number. The right-hand side of Eq. (3.47) represents the exact transport operator, a pure imaginary number, and $\xi_{\theta,\phi}$ is a pure real number. We now consider the semi-discrete counterpart to Eq. (3.47). Taking the spatial Fourier transform of the semi-discrete system (3.7) applied to the linear problem (3.38), we obtain:

$$\hat{w}_t = \hat{\mathscr{C}}(\boldsymbol{\xi}, \dot{\mathbf{A}})\frac{\hat{w}}{\Delta t} \tag{3.48}$$

with $\hat{\mathscr{C}}$ the Fourier symbol of the spatial operator. Introducing the modified wave number:

$$\xi^*_{\theta,\phi} = i\,\frac{\hat{\mathscr{C}}(\boldsymbol{\xi}, \dot{\mathbf{A}})}{|\dot{\mathbf{A}}|} \tag{3.49}$$

Eq. (3.48) can be rewritten as:

$$\hat{w}_t = -i\,|\dot{\mathbf{A}}|\xi^*_{\theta,\phi}\frac{\hat{w}}{\Delta t} \tag{3.50}$$

which represents the numerical counterpart of the exact operator (3.47). Equation (3.49) is called the approximate dispersion relation of the semi-discrete scheme, and relates a given reduced wave number $\xi_{\theta,\phi}$ to its numerical representation $\xi^*_{\theta,\phi}$ which is a complex number. If the imaginary part of $\xi^*_{\theta,\phi}$ is negative, then any Fourier mode set as an initial condition to (3.50) cannot grow without bound. Precisely, the spatial discretization operator $\mathscr{C}$ is Cauchy-stable if:

$$\forall \dot{\mathbf{A}} \in \mathbb{R}^3, \quad \sup_{\boldsymbol{\xi}\in[-\pi,\pi]^3} \Im\mathrm{m}(\xi^*_{\theta,\phi}) \leqslant 0 \tag{3.51}$$

Any spatial discretization satisfying the stability condition (3.51) leads to an unconditionally stable fully discrete scheme whenever it is combined to an A-stable

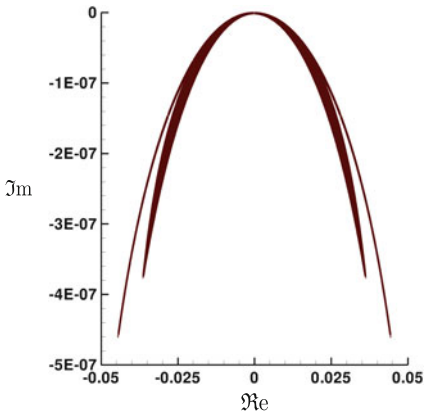


Fig. 3.5 Representation in the complex plane of the Cauchy-stability criterion (3.51) for RBC*i*

time integration method. The expression of $\hat{\mathscr{C}}$ in the case of RBC*i* discretization is determined in 3D as in [4] for RBC3 in 2D. Given the complexity of the analytical expression of $\xi^*_{\theta,\phi}$ for RBC*i*, a computer code has been written to study its mathematical properties. We numerically check that RBC*i* is Cauchy stable. To this purpose, the imaginary part of $\xi^*_{\theta,\phi}$ is computed for CFL numbers $\dot{A}$, $\dot{B}$ and $\dot{C}$ ranging in $[-2, 2]$ with a step $\Delta\dot{A} = \Delta\dot{B} = \Delta\dot{C} = 1/100$ and the wave vector $\boldsymbol{\xi}$ ranging in $[-\pi, \pi]^2 \times [0, \pi]$ with steps $\Delta\xi = \Delta\eta = \Delta\zeta = \pi/50$. Figure 3.5 displays criterion (3.51) for RBC*i*. Precisely, it shows that RBC*i* verifies condition (3.51) for any CFL and any wave number, which shows that RBC*i* is a Cauchy-stable approximation.

3.7 Numerical Applications

For the following computations, the time derivative is approximated by an A-stable second-order accurate Backward Linear Multistep Method [2]. Other time discretizations would be possible (e.g. Runge–Kutta methods), but their resolution would be very costly with the present formulation of RBC schemes, because of the complex spatial operator applied to w_t. Improvements are possible, but they are beyond the scope of this paper. A second-order discretization of the viscous terms is adopted.

3.7.1 Advection of an Inviscid Vortex

The convergence order of FV RBC schemes is investigated for the advection of an inviscid vortex flow problem. In this test case, an isentropic vortex is added to a uniform freestream flow. The free stream density and pressure are given by

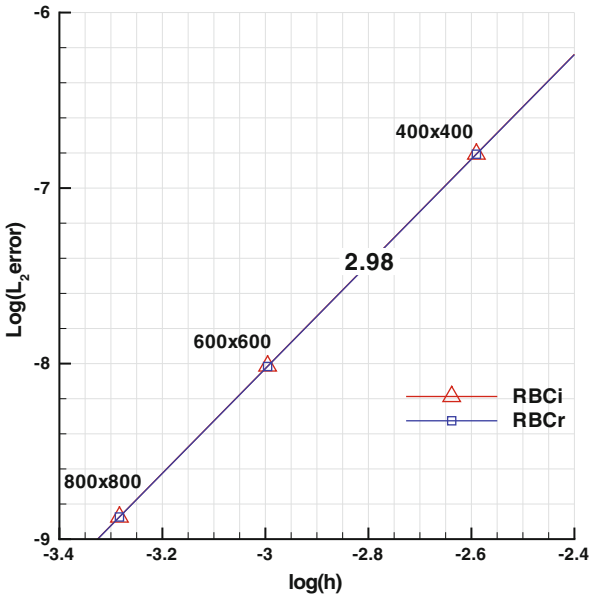


Fig. 3.6 L_2 error on ρ for Cartesian meshes

$\rho_\infty = 1$, $p_\infty = 1/\gamma$ with $\gamma = 1.4$ the ratio of specific heats. The freestream flow velocity components are chosen such that $u_\infty = v_\infty = \sqrt{2}$ in order to yield an advection at supersonic speed along the grid diagonal. The problem is therefore fully multidimensional. The perturbation values of the vortex are those given in [15]. The vortex evolution is computed up to $t_f = 14$. The computational domain extends from -15 to $+15$ in both directions with periodic boundary conditions. The initial position of the core vortex is $x_{c0} = y_{c0} = -10$. The convergence study is first done on a series of Cartesian regular grids with 400×400, 600×600 and 800×800 cells. Since the time approximation is second-order accurate, we choose the time step such that $\Delta t \propto h^{\frac{p}{2}}$, where $p \geqslant 2$ is the spatial order of accuracy and h the spatial mesh size, in order to recover a global order of accuracy equal to p. Figure 3.6 plots the L_2 norm of the error with respect to the exact solution (for the density field) as a function of the mesh size, in a double-log scale. A slope almost equal to 3 is recovered both for RBCr and RBCi, which confirms the preceding theoretical results. In order to introduce mesh irregularities we randomly shake the meshes by an amount σ. In practice, each cell node is randomly moved within a circle centered on its initial position and of radius σh where h is the mesh size for a regular Cartesian mesh with the same number of nodes. Figure 3.7 shows that for mildly deformed grids ($\sigma = 0.05$) RBCi remains third order accurate, unlike RBCr, for which a lower slope (2.85) is found. For a more severly deformed grid ($\sigma = 0.1$) neither RBCr nor RBCi are third order accurate any more. Nevertheless, RBCi is still second order accurate, in agreement with the theoretical analysis of Sect. 3.5, whereas RBCr is less than second order (see Fig. 3.8).

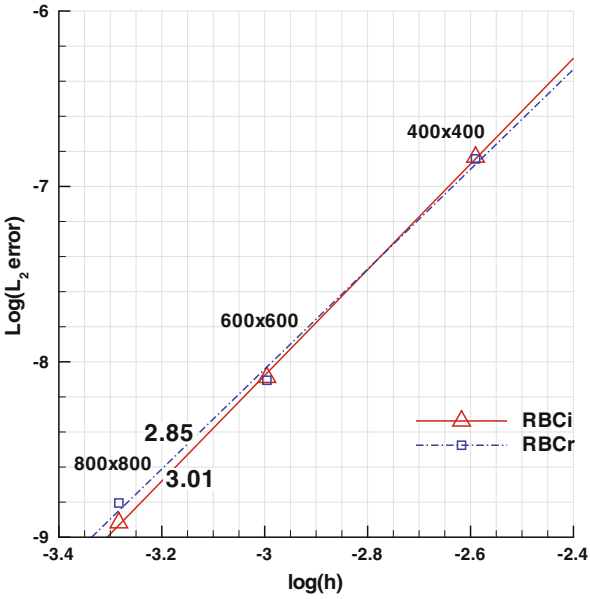


Fig. 3.7 L_2 error on ρ for shaken meshes with $\sigma = 0.05$

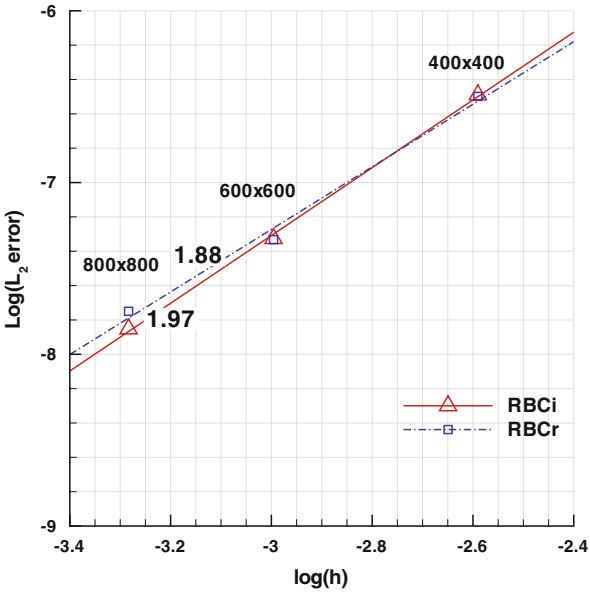


Fig. 3.8 L_2 error on ρ for shaken meshes with $\sigma = 0.1$

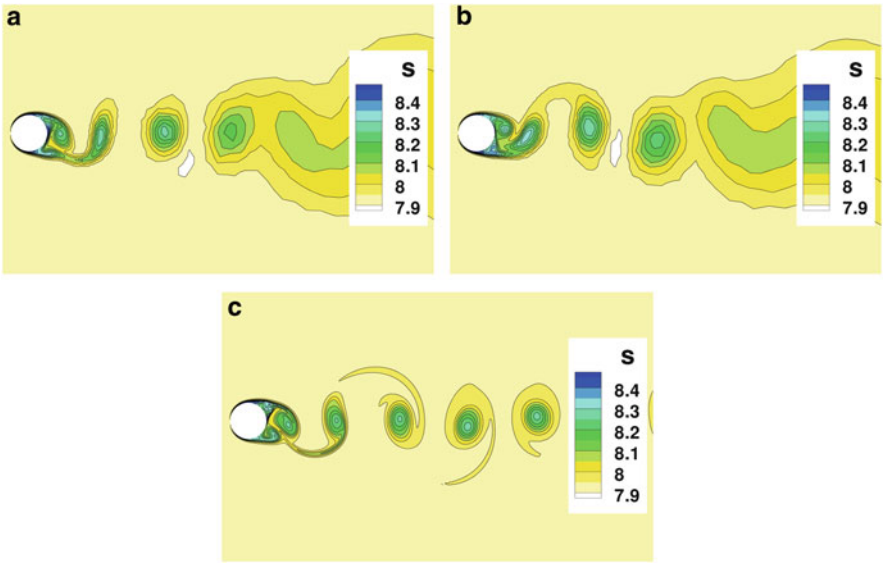


Fig. 3.9 Snapshot of the entropy field. (**a**) RBCi, 100×168 shaken mesh, $\sigma = 0.4$. (**b**) RBCr, 100×168 shaken mesh, $\sigma = 0.4$. (**c**) RBC5, reference, 401×673 regular polar mesh

3.7.2 *Laminar Flow Past a Cylinder*

We consider an unsteady laminar flow past a two-dimensional circular cylinder at a Reynolds number of 1,200 and a Mach number of 0.3. The computational grid is a regular polar mesh composed by 100×168 cells. The grid points are uniformly distributed in the radial direction and clustered close to the cylinder wall in the normal direction. The external boundary is located 20 diameters far from the cylinder and the distance between the wall and the first grid point is 0.0085 times the diameter. The adimensional time step used is $\Delta t = 3.125 \times 10^{-3}$. A reference computation has been carried-out on a fine mesh composed by 400×672 cells, using a nominally 5th-order accurate RBC scheme [10]. In addition to the regular mesh, a randomly deformed meshe is generated by shaking the nodes of the regular one by a random amount $\sigma = 0.4$. An overall view of the numerical solutions of RBCr and RBCi on the most deformed grid is provided in Fig. 3.9 along with the reference solution. Even if the vortex street is dissipated more quickly (because of the mesh coarsening and deformation), RBCi provides results in somewhat closer agreement with the reference than RBCr. For a more quantitative comparison, we computed discrete Fourier transforms of the unsteady vertical force acting on the cylinder for different schemes and meshes. The frequency content of the signal is essentially represented by a single mode, associated to vortex shedding. This is used to compute the Strouhal number St (based on free-stream velocity and cylinder diameter) of the vortex street. The reference computation returns $St = 0.25$. On the shaken grid RBCi returns a Strouhal number equal to 0.24, whereas RBCr gives $St = 0.23$.

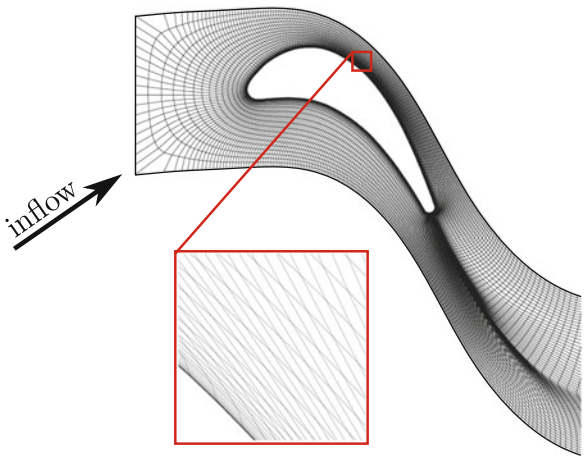


Fig. 3.10 VKI LS-59 turbine blade, computational grid

3.7.3 VKI LS-59 Turbine Cascade

The final application is the simulation of a transonic turbulent flow through a turbine cascade. The VKI LS-59 is a high-loaded rotor blade with a thick, rounded trailing edge originally designed for near-sonic exit flow conditions. This rotor blade has been extensively tested in various European wind tunnels and computed by several authors, see e.g. [6, 7]. The flow conditions considered correspond to a unit exit isentropic Mach number and a Reynolds number of 7.44×10^5 based on the chord length and mean outlet velocity. The inlet flow angle is equal to 30°. Unsteady two-dimensional computations are carried out using a C-grid of 384×32 cells with a conformal join along the branch cut (see Fig. 3.10). The enforcing of a conformal join leads to a highly distorted mesh in the suction side region for this high-deflection blade. The non-dimensional time step is selected in order to get approximately 30 time steps per shedding cycle. The computations are performed using the Spalart-Allmaras turbulence model. To investigate the importance of taking into account mesh irregularities for highly distorted grids RBCr and RBCi are compared to each other. Snapshots of the unsteady entropy field computed with RBCr and RBCi are provided in Fig. 3.11, showing periodical vortex shedding. Note that the second order RBC scheme gives a steady solution. Nevertheless, the solution of RBCr exhibits non-physical features on the suction side region of the blade where the mesh is highly distorted, cf. Fig. 3.11b. The solution is indeed polluted by significant spurious numerical oscillations originating from the regions of the computational domain where the grid is more severely distorted, cf. Fig. 3.11b. Figure 3.12 provides the Fourier spectra of the blade load (tangential force acting on the blade) for RBCi and RBCr. The computed Strouhal numbers are based on the main frequency, the isentropic outlet velocity and the thickness of the trailing edge (equal to 0.06 the axial chord for this blade). RBCi scheme returns a value of 0.24 close to the experimentally measured range [7] (0.2, 0.4), along with

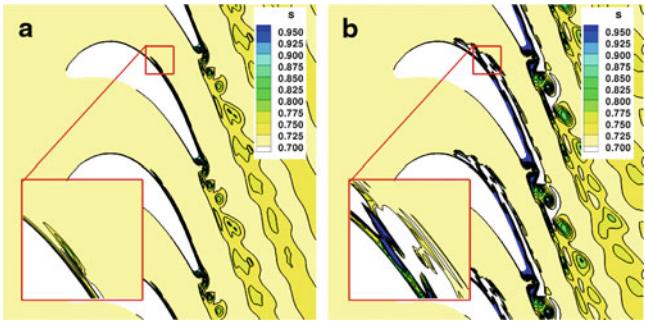


Fig. 3.11 Snapshot of the unsteady entropy field. (**a**) RBCi. (**b**) RBCr

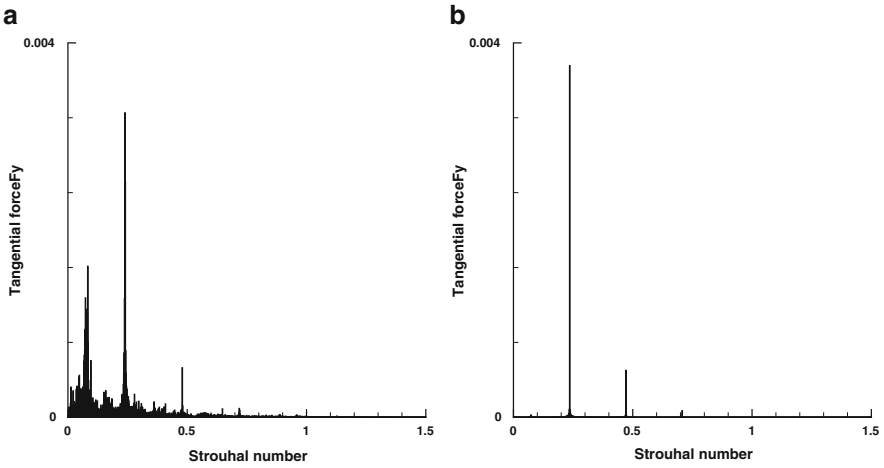


Fig. 3.12 Fourier spectra of the unsteady tangential force on the blade. (**a**) RBCr. (**b**) RBCi

its harmonics. The RBCr scheme returns the same Strouhal value associated to the main frequency along with a rich frequency spectrum. These additional frequencies are probably due to numerical errors introduced when mesh irregularities are not taken into account.

3.8 Conclusions

A third-order finite volume Residual-Based Compact scheme, RBCi, has been designed for the numerical simulation of compressible flows on structured meshes. The scheme takes into account mesh deformations, and it is third-order on mildly distorted structured grids and second-order on highly distorted meshes. An analysis of the Fourier symbol of the proposed spatial approximation demonstrates that the

resulting discretization is dissipative and Cauchy-stable for all flow configurations. Numerical tests confirm the theoretical results and demonstrate the importance of using a weighted scheme formulation on very irregular meshes. RBCi is more accurate and robust than RBCr when highly deformed grids are used. This enables its application to complex configurations of industrial interest, like turbomachinery.

In terms of numerical cost, RBCi is more expensive than RBCr, essentially because of the calculation of weighed operator coefficients: the CPU time of an RBCi computation is roughly 1.27 times the time of an RBCr one. Nevertheless, RBCi ensures high accuracy and robustness on computational grids of poor quality, which enables reducing time required for mesh generation. For all of these reasons, RBCi appears to be a good candidate for high-order industrial CFD.

Acknowledgements This research has been done within the framework of the European project IDIHOM (Industrialization of High Order Methods) which aims to promote the use of high-order numerical methods by the European aerospace industry.

References

1. elsA. http://elsa.onera.fr
2. Dahlquist, G.: A special stability problem for linear multistep methods. BIT **3**, 27–43 (1963)
3. Du, X., Corre, C., Lerat, A.: A third-order finite-volume residual-based scheme for the 2D Euler equations on unstructured grids. J. Comput. Phys. **230**, 4201–4215 (2011)
4. Grimich, K., Cinnella, P., Lerat, A.: Spectral properties of high-order residual-based compact schemes for unsteady compressible flows. J. Comput. Phys. **252**, 142–162 (2013)
5. Hanss, G.: Schémas numériques compacts basés sur le résidu en maillage irrégulier pour les équations de Navier-Stokes en compressible. PhD thesis, Arts et Métiers ParisTech (2002)
6. Inoue, M., Furukawa, M.: Numerical Methods for Flow Calculations in Turbomachines. VKI Lecture Series 1994–2006. VKI, Rhode-Saint-Genèse (1994)
7. Kiock, R., Lehthaus, F., Baines, N.C., Sieverding, C.H.: The transonic flow through a plane turbine cascade as measured in four European wind tunnels. J. Eng. Gas Turbines Power **108**, 277–285 (1986)
8. Lerat, A., Corre, C.: A residual-based compact scheme for the compressible Navier-Stokes equations. J. Comput. Phys. **170**, 642–675 (2001)
9. Lerat, A., Corre, C.: Residual-based compact schemes for multidimensional hyperbolic systems of conservation laws. Comput. Fluids **31**, 639–661 (2002)
10. Lerat, A., Grimich, K., Cinnella, P.: On the design of high order residual-based dissipation for unsteady compressible flows. J. Comput. Phys. **235**, 32–51 (2013)
11. MacCormack, R.W., Paullay, A.J.: Compuational effinciency achieved by time-splitting of finite-difference operators. AIAA PAPER 72–154 (1972)
12. Marsden, O., Bogey, C., Bailly, C.: High-order curvilinear simulations of flows around non-Cartesian bodies. J. Comput. Acoust. **13**, 731–748 (2005)
13. Rezgui, A., Cinnella, P., Lerat, A.: Third-order finite volume schemes for Euler computations on curvilinear meshes. Comput. Fluids **30**, 875–901 (2001)
14. Visbal, M., Gaitonde, D.V.: Compact finite difference schemes on non-uniform meshes. Application to direct numerical simulation of compressible flows. AIAA J. **37**, 1231–1239 (1999)
15. Yee, H.C., Vinokur, M., Djomehri, M.J.: Entropy splitting and numerical dissipation. J. Comput. Phys. **162**, 33–81 (2000)

Chapter 4
Parallel Implementation of k-Exact Finite Volume Reconstruction on Unstructured Grids

Florian Haider, Pierre Brenner, Bernard Courbet, and Jean-Pierre Croisille

Abstract We present a high-order piecewise polynomial approximation for finite volume schemes. This method is designed for general unstructured polyhedral grids. A new algorithm for the calculation of the approximants is discussed. The main geometric property of the procedure is that the direct neighborhood only is needed. This allows for an efficient parallel implementation.

4.1 Introduction

An important contribution to the design of higher-order versions of the upwind finite-volume method for gas dynamics is the one by Barth and Frederickson [4]. This suggested a high-order finite volume algorithm to approximate Euler's equations on general unstructured grids. The time-marching of their finite volume method for the Euler equations can be decomposed into three steps:

1. Step 1: high-order reconstruction in each cell,
2. Step 2: high-order numerical integration of the numerical flux,
3. Step 3: high-order time stepping.

F. Haider • B. Courbet
ONERA, BP72 – 92322 Châtillon-sous-Bagneux, France
e-mail: florian.haider@onera.fr; bernard.courbet@onera.fr

P. Brenner
ASTRIUM Space Transportation – Aerodynamics, BP3002 – 78133 Les Mureaux, France
e-mail: pierre.brenner@astrium.eads.net

J.-P. Croisille (✉)
Département de Mathématiques, Institut Elie Cartan de Lorraine, Université de Lorraine, UMR CNRS 7502, Ile du Saulcy, 57045 Metz, France
e-mail: jean-pierre.croisille@univ-lorraine.fr

R. Abgrall et al. (eds.), *High Order Nonlinear Numerical Schemes for Evolutionary PDEs*, Lecture Notes in Computational Science and Engineering 99,
DOI 10.1007/978-3-319-05455-1_4,

Selecting efficient interpolants to properly handle Step 1 has been the center of attention in the CFD community for more than 20 years. This quest is motivated by the increasing need of industrial applications such as jet noise prediction, LES modeling, internal flow simulation in aerothermochemistry. Efficient simulations of such problems clearly require approximations beyond second order.

As is well known, high-order interpolation schemes for advective problems can be unstable [1]. This question was extensively studied in the framework of adaptive stencil selection (ENO/WENO schemes [12]). Other approaches include non polynomial functions such as radial basis functions (RBF) or splines [10].

The present contribution is the continuation of previous studies devoted to designing high-order centered polynomial reconstructions [7–9].[1]

The reconstruction step 1 above clearly belongs to the theory of approximation. It can be stated as follows: consider a smooth function $u : \Omega \subset \mathbb{R}^d \to \mathbb{R}$ supposed to be known by its cell averages on an unstructured grid. We seek a piecewise polynomial (possibly discontinuous) approximant w to the function u. The local accuracy in each cell is subject to the *k-exact* condition: the approximant procedure $u \mapsto w$ is said to be $(k+1)$-*order accurate* [3–5] if any locally polynomial function of degree k is invariant.

An important question when dealing with k-exact interpolants ($k > 1$) is the following. On the one hand computational efficiency requires accessing data close to the cell (typically the Von-Neumann neighborhood, see (4.3)). On the other hand high-order approximations clearly require accessing data (cell averages) belonging to a wide stencil. Note that the question of designing efficient reconstruction algorithms for ENO schemes on triangular grids is addressed in [2].

The main outcome of this paper is an algorithm to efficiently calculate k-exact reconstructions from non local data in an efficient way.

The outline of the paper is as follows: in Sect. 4.2 we briefly review notation for finite-volume approximations on general grids as well as elements of tensor algebra useful in the sequel. In Sect. 4.3 we restate and examine rigorously the approximation method of k-exact polynomial reconstruction. Our algorithm, introduced in [7, 8] is expounded in Sect. 4.4. Finally we present in Sects. 4.5 and 4.6 numerical results showing the relevance of the preceding analysis.

Note that we do not address in this paper how to handle non smooth data, which is clearly an essential aspect when dealing with high-order approximation methods in the CFD context. This is the subject of ongoing work.

[1] The present study takes place within several industrial projects: the CEDRE and FLUSEPA projects are multiphysics codes respectively developed at ONERA/DSNA (Châtillon-sous-Bagneux, FRANCE) and at ASTRIUM Space Transportation (Les Mureaux, FRANCE) for applications in aerothermochemistry and propulsion.

4.2 Notation

4.2.1 Finite-Volume Notation

We consider a computational domain $\Omega \subset \mathbb{R}^d$ and a *finite-volume grid* of Ω made of polyhedral (compact) cells $\mathscr{T}_\alpha$.[2] The barycenter of $\mathscr{T}_\alpha$ is denoted $\boldsymbol{x}_\alpha$ and $|\mathscr{T}_\alpha|$ is its d-volume. The *internal face* $\mathscr{A}_{\alpha\beta} = \mathscr{T}_\alpha \cap \mathscr{T}_\beta$ has a $(d-1)$-volume $\left|\mathscr{A}_{\alpha\beta}\right|$ and a barycenter $\boldsymbol{x}_{\alpha\beta}$. The surface vector of the face $\mathscr{A}_{\alpha\beta}$ is $\boldsymbol{a}_{\alpha\beta}$ with orientation from cell $\mathscr{T}_\alpha$ to $\mathscr{T}_\beta$. The local geometry nearby the cell $\mathscr{T}_\alpha$ is described using the two vectors $\boldsymbol{h}_{\alpha\beta}$ and $\boldsymbol{k}_{\alpha\beta}$ defined by (Fig. 4.1)

$$\boldsymbol{h}_{\alpha\beta} \triangleq \boldsymbol{x}_\beta - \boldsymbol{x}_\alpha \text{ ; for all cells } \mathscr{T}_\alpha, \mathscr{T}_\beta, \tag{4.1}$$

$$\boldsymbol{k}_{\alpha\beta} \triangleq \boldsymbol{x}_{\alpha\beta} - \boldsymbol{x}_\alpha \text{ ; for all adjacent cells } \mathscr{T}_\alpha, \mathscr{T}_\beta \,. \tag{4.2}$$

The nth *grid neighborhood* of the cell $\mathscr{T}_\alpha$ is defined as follows. First we define the *first grid neighborhood* (or *Von Neumann neighborhood*) by

$$\mathbb{V}_\alpha^{(1)} \triangleq \left\{\beta |\, \mathscr{T}_\beta \text{ is adjacent to } \mathscr{T}_\alpha\right\} \cup \{\alpha\}\,, \tag{4.3}$$

then the nth *grid neighborhood* is recursively defined by

$$\mathbb{V}_\alpha^{(n)} \triangleq \bigcup_{\gamma \in \mathbb{V}_\alpha^{(n-1)}} \mathbb{V}_\gamma^{(1)}\,. \tag{4.4}$$

On a finite-volume grid $\cup_{1\le\alpha\le N} \mathscr{T}_\alpha$ of size N we consider *grid functions* $\mathfrak{u}$. In practice a grid function is given by the values u_i,

$$\mathfrak{u} \triangleq (u_1, \ldots, u_N) \in \mathbb{R}^N\,. \tag{4.5}$$

To any grid function $\mathfrak{u}$ we further associate the *piecewise polynomial reconstruction* defined by

$$\boldsymbol{x} \mapsto w\left[\mathfrak{u}\right](\boldsymbol{x}) = \sum_{\alpha=1}^{N} w_\alpha\left[\mathfrak{u}\right](\boldsymbol{x})\, \chi_{\mathscr{T}_\alpha}(\boldsymbol{x})\,, \tag{4.6}$$

where $\chi_{\mathscr{T}_\alpha}$ is the characteristic function of the cell $\mathscr{T}_\alpha$ and where the $w_\alpha\left[\mathfrak{u}\right](\boldsymbol{x})$ denote polynomial functions of the variable $\boldsymbol{x}$. The particular case $w_\alpha\left[\mathfrak{u}\right] = u_\alpha$ corresponds to the piecewise constant function associated with $\mathfrak{u}$ by the relation

[2] $\mathscr{T}_\alpha$ is not required to be convex, but the barycenter $\boldsymbol{x}_\alpha$ must be located in the topological interior of $\mathscr{T}_\alpha$.

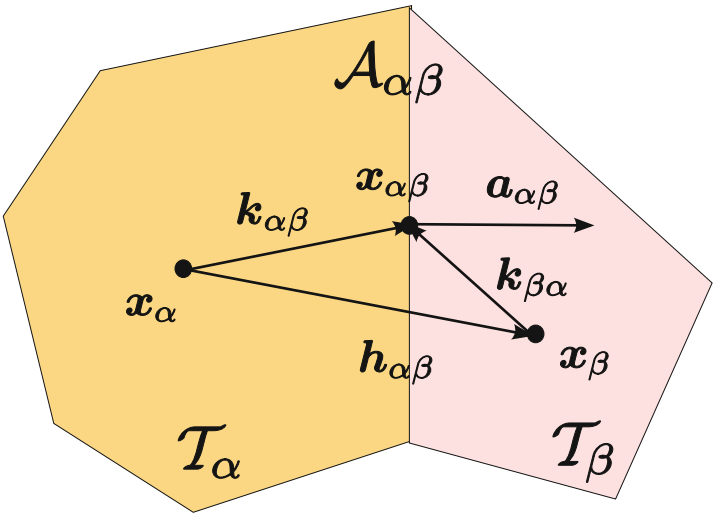


Fig. 4.1 Two cells $\mathscr{T}_\alpha$ and $\mathscr{T}_\beta$ sharing a common face $\mathscr{A}_{\alpha\beta}$ in a two-dimensional polyhedral grid. The barycenters of $\mathscr{T}_\alpha$ and $\mathscr{T}_\beta$ are $\boldsymbol{x}_\alpha$ and $\boldsymbol{x}_\beta$, respectively. The face $\mathscr{A}_{\alpha\beta}$ has a barycenter $\boldsymbol{x}_{\alpha\beta}$ and a surface vector $\boldsymbol{a}_{\alpha\beta}$

$$\boldsymbol{x} \mapsto w^0\left[\mathfrak{u}\right](\boldsymbol{x}) = \sum_{\alpha=1}^{N} u_\alpha \chi_{\mathscr{T}_\alpha}(\boldsymbol{x})\,. \tag{4.7}$$

In analogy to the finite difference setting the consistency analysis of a reconstruction is based upon analyzing how it operates on a given function $u(\boldsymbol{x})$ of the continuous variable $\boldsymbol{x}$: we associate with the function $u(\boldsymbol{x})$ its matching gridfunction $\overline{\mathfrak{u}} = (\overline{u}_1, \ldots, \overline{u}_N)$ where $\overline{u}_\alpha$ is defined by

$$\overline{u}_\alpha \triangleq \frac{1}{|\mathscr{T}_\alpha|}\int_{\mathscr{T}_\alpha} u(\boldsymbol{x})\, dx\,. \tag{4.8}$$

For each $\alpha = 1, \ldots N$ the operator $\mathfrak{R}_\alpha$ is given by

$$\mathfrak{R}_\alpha : \mathbb{R}^N \ni \mathfrak{u} \longmapsto w_\alpha\left[\mathfrak{u}\right] \in \mathbb{P}_k(\mathbb{R}^d)\,. \tag{4.9}$$

Two important properties of the reconstruction (4.6) are the following:

- The reconstruction is *linear* if each operator $\mathfrak{R}_\alpha$ in (4.9) is linear as a function of the argument $\mathfrak{u}$.
- The reconstruction is *conservative* if the operator $\mathfrak{R}_\alpha$ satisfies the property

$$\frac{1}{|\mathscr{T}_\alpha|}\int_{\mathscr{T}_\alpha} \boldsymbol{w}_\alpha\left[\overline{\mathfrak{u}}\right](\boldsymbol{x})\, d\boldsymbol{x} = \overline{u}_\alpha \text{ for all } \overline{\mathfrak{u}} = (\overline{u}_1, \ldots, \overline{u}_N) \in \mathbb{R}^N\,. \tag{4.10}$$

In terms of gridfunctions, the condition (4.10) can be expressed as

$$\overline{\boldsymbol{w}_\alpha\,[\overline{\mathrm{u}}]} = \overline{\mathrm{u}}. \tag{4.11}$$

For physical reasons, condition (4.10) is crucial when using piecewise polynomial reconstruction in the context of finite volume schemes for conservation laws: obviously the total density, momentum and energy must be preserved through all interpolations procedures.

4.2.2 Tensor Notation

In order to handle higher-order derivatives within the multidimensional geometrical setting it appeared convenient to rely on some classical *tensor analysis*. For example to each pair of cells $\mathscr{T}_\alpha$ and $\mathscr{T}_\beta$ we will need the momentum $\boldsymbol{z}^{(k)}_{\alpha\beta}$ defined by

$$\boldsymbol{z}^{(k)}_{\alpha\beta} \triangleq \frac{1}{|\mathscr{T}_\beta|}\int_{\mathscr{T}_\beta} (\boldsymbol{x}-\boldsymbol{x}_\alpha)^{\otimes k}\, dx. \tag{4.12}$$

Considering $\boldsymbol{z}^{(k)}_{\alpha\beta}$ as an element of the space $\mathrm{S}^m\left(\mathbb{R}^d\right)$ of symmetric tensors of rank m greatly facilitates calculations of such momenta.

Recall that a symmetric tensor $\boldsymbol{a} \in \mathrm{S}^m\left(\mathbb{R}^d\right)$ has components $a_{i_1\cdots i_m}$ where $a_{i_1\cdots i_m} = a_{i_{\pi(1)}\cdots i_{\pi(m)}}$ for all permutations π. Thus each tensor $\boldsymbol{a} \in \mathrm{S}^m\left(\mathbb{R}^d\right)$ is specified by $\binom{m+d-1}{m}$ independent components. For $\boldsymbol{a}, \boldsymbol{b} \in \mathrm{S}^m\left(\mathbb{R}^d\right)$ the contracted product is defined by $\boldsymbol{a} \bullet \boldsymbol{b} \triangleq \sum_{i_1=1}^d \cdots \sum_{i_m=1}^d a_{i_1\cdots i_m} b_{i_1\cdots i_m}$.

The components of the tensor $\boldsymbol{z}^{(k)}_{\alpha\beta}$ in (4.12) are

$$z^{(k)}_{\alpha\beta;i_1,\ldots,i_k} \triangleq \frac{1}{|\mathscr{T}_\beta|}\int_{\mathscr{T}_\beta} (x_{i_1}-x_{\alpha;i_1})\cdots\left(x_{i_k}-x_{\alpha;i_k}\right)\, dx\,. \tag{4.13}$$

When $\beta = \alpha$, (4.12) reduces to the kth momentum of the cell $\mathscr{T}_\alpha$ namely

$$\boldsymbol{x}^{(k)}_\alpha \triangleq \boldsymbol{z}^{(k)}_{\alpha\alpha} = \frac{1}{|\mathscr{T}_\alpha|}\int_{\mathscr{T}_\alpha} (\boldsymbol{x}-\boldsymbol{x}_\alpha)^{\otimes k}\, dx\,. \tag{4.14}$$

In index notation (4.14) is expressed as

$$x^{(k)}_{\alpha;i_1,\ldots,i_k} \triangleq z^{(k)}_{\alpha\alpha;i_1,\ldots,i_k} = \frac{1}{|\mathscr{T}_\alpha|}\int_{\mathscr{T}_\alpha} (x_{i_1}-x_{\alpha;i_1})\cdots\left(x_{i_k}-x_{\alpha;i_k}\right)\, dx\,. \tag{4.15}$$

Note that when $k = 0, 1$, we have

$$z_{\alpha\beta}^{(0)} = 1, \quad z_{\alpha\beta}^{(1)} = \boldsymbol{h}_{\alpha\beta} \tag{4.16}$$

and

$$\boldsymbol{x}_{\alpha}^{(0)} = 1, \quad \boldsymbol{x}_{\alpha}^{(1)} = \boldsymbol{h}_{\alpha\alpha} = \boldsymbol{x}_{\alpha} - \boldsymbol{x}_{\alpha} = 0. \tag{4.17}$$

4.3 Algebra of k-Exact Reconstruction for Derivatives

This section is devoted to the general setting of k-exact reconstructions [4] for $k \geq 1$. A function u is called *k-exact* on the grid neighborhood $\mathbb{W}_{\alpha}$ if the function u restricted to $\mathbb{W}_{\alpha}$ is polynomial. Otherwise stated, there exists a polynomial function $p \in \mathbb{P}_k(\mathbb{R}^d)$ such that the restriction of u and p to $\mathscr{T}(\mathbb{W}_{\alpha})$ coincide

$$u|_{\mathscr{T}(\mathbb{W}_{\alpha})} = p|_{\mathscr{T}(\mathbb{W}_{\alpha})} . \tag{4.18}$$

So the k-exact property (4.18) can be expressed as follows: the operator $\mathfrak{R}_{\alpha} : \mathbb{R}^N \to \mathbb{P}_k(\mathbb{R}^d)$ is *k-exact* on the grid neighborhood $\mathbb{W}_{\alpha}$ if the local approximant $w_{\alpha}[\overline{\mathrm{u}}]$ reproduces all k-exact functions u on $\mathbb{W}_{\alpha}$,

$$w_{\alpha}[\overline{\mathrm{u}}]|_{\mathscr{T}(\mathbb{W}_{\alpha})} = u|_{\mathscr{T}(\mathbb{W}_{\alpha})} . \tag{4.19}$$

Expressing the polynomial function $p \in \mathbb{P}_k(\mathbb{R}^d)$ in terms of its successive derivatives at $\boldsymbol{x}_{\alpha}$ yields

$$p(\boldsymbol{x}) = p(\boldsymbol{x}_{\alpha}) + \sum_{j=1}^{k} \frac{1}{j!} \sum_{i_1 \cdots i_j = 1}^{d} \left. \frac{\partial p}{\partial x_{i_1} \cdots \partial x_{i_j}} \right|_{\boldsymbol{x} = \boldsymbol{x}_{\alpha}} (x_{i_1} - x_{\alpha;i_1}) \cdots \left(x_{i_j} - x_{\alpha;i_j}\right). \tag{4.20}$$

It is therefore the case that any k-exact reconstruction of the polynomial $p \in \mathbb{P}_k(\mathbb{R}^d)$ is obtained by reconstructing the mth derivatives of the polynomial p for $0 \leq m \leq k$ from its cell averages. Observe that the mth derivative $D^{(m)}v(x)$ of a given smooth function v can be considered as a symmetric tensor, element of the set $\mathrm{S}^m\left(\mathbb{R}^d\right)$. Our algebraic setting for k-exact reconstruction can be further expressed as follows:

Definition 4.3.1 (k-exact reconstruction operator). Let $\mathbb{W}_{\alpha}$ be a grid neighborhood of cell $\mathscr{T}_{\alpha}$ and let $k \in \mathbb{N}$ with $m \leq k$. A ***k-exact*** mth *derivative on* $\mathbb{W}_{\alpha}$ is a *linear map* $\boldsymbol{w}_{\alpha}^{(m|k)} : \mathbb{R}^N \longrightarrow \mathrm{S}^m\left(\mathbb{R}^d\right)$ given by coefficients $w_{\alpha\beta;i_1\ldots i_j}^{(m|k)}$ is such that for all functions v that are k-exact on $\mathbb{W}_{\alpha}$

$$\sum_{\beta \in \mathbb{W}_{\alpha}} w_{\alpha\beta;i_1\ldots i_m}^{(m|k)} \overline{v}_{\beta} = \left. \frac{\partial v}{\partial x_{i_1} \cdots \partial x_{i_m}} \right|_{\boldsymbol{x} = \boldsymbol{x}_{\alpha}} . \tag{4.21}$$

Furthermore the *reconstruction stencil* of $\boldsymbol{w}_\alpha^{(m|k)}$ at cell $\mathscr{T}_\alpha$ is defined as the set of indices $\beta \in \{1, \ldots, N\}$ such that $w_{\alpha\beta;i_1\ldots i_j}^{(m|k)} \neq 0$.

A k-exact polynomial reconstruction at cell $\mathscr{T}_\alpha$ is thus equivalent to a set of k-exact mth derivatives $\boldsymbol{w}_\alpha^{(m|k)}$ at cell $\mathscr{T}_\alpha$ for $0 \leq m \leq k$. The constant terms $w_{\alpha\beta}^{(0|k)}$ are uniquely determined by Eq. (4.10). Therefore, the reconstructed polynomial in cell $\mathscr{T}_\alpha$ has the general form

$$w_\alpha\left[\overline{u}\right](\boldsymbol{x}) = \overline{u}_\alpha + \sum_{m=1}^{k} \frac{1}{m!} \sum_{i_1\cdots i_m} \sum_{\beta\in\mathbb{W}_\alpha} w_{\alpha\beta;i_1\cdots i_m}^{(m|k)} \overline{u}_\beta \left[(x_{i_1} - x_{\alpha;i_1})\cdots(x_{i_m} - x_{\alpha;i_m}) - x_{\alpha;i_1\cdots i_m}^{(m)}\right]. \tag{4.22}$$

Using (4.12) the cell averages $\overline{p}_\gamma$ on a cell $\mathscr{T}_\gamma$ of the function $p(x)$ in (4.20) can be expressed as

$$\overline{p}_\gamma = p(\boldsymbol{x}_\alpha) + \sum_{j=1}^{k} \frac{1}{j!} D^{(j)} p\Big|_{\boldsymbol{x}_\alpha} \bullet \boldsymbol{z}_{\alpha\gamma}^{(j)} \tag{4.23}$$

or equivalently

$$\overline{p}_\gamma = p(\boldsymbol{x}_\alpha) + \sum_{j=1}^{k} \frac{1}{j!} \sum_{i_1\cdots i_j=1}^{d} z_{\alpha\beta;i_1\ldots i_j}^{(j)} \frac{\partial p}{\partial x_{i_1}\cdots\partial x_{i_j}}\bigg|_{\boldsymbol{x}=\boldsymbol{x}_\alpha}. \tag{4.24}$$

Using (4.24), the reconstruction error of a k-exact mth derivative $\boldsymbol{w}_\alpha^{(m|k)}$ on a grid neighborhood $\mathbb{W}_\alpha$ at cell $\mathscr{T}_\alpha$ applied to a j-exact function $u(x)$ (with $j > k$) can be expressed as

$$\sum_{\beta\in\mathbb{W}_\alpha} \boldsymbol{w}_{\alpha\beta}^{(m|k)} \overline{u}_\beta - D^{(m)} u\Big|_{\boldsymbol{x}=\boldsymbol{x}_\alpha} = \sum_{l=k+1}^{j} \frac{1}{l!} \sum_{\beta\in\mathbb{W}_\alpha} \boldsymbol{w}_{\alpha\beta}^{(m|k)} \left(\boldsymbol{z}_{\alpha\beta}^{(l)} \bullet D^{(l)} u\Big|_{\boldsymbol{x}=\boldsymbol{x}_\alpha}\right). \tag{4.25}$$

Equation (4.25) can be expressed using index notation as

$$\sum_{\beta\in\mathbb{W}_\alpha} w_{\alpha\beta;i_1\ldots i_m}^{(m|k)} \overline{u}_\beta - \frac{\partial u}{\partial x_{i_1}\cdots\partial x_{i_m}}\bigg|_{\boldsymbol{x}=\boldsymbol{x}_\alpha} = \sum_{l=k+1}^{j} \frac{1}{l!} \sum_{\beta\in\mathbb{W}_\alpha} w_{\alpha\beta;i_1\ldots i_m}^{(m|k)} \sum_{i_1\cdots i_j=1}^{d} z_{\alpha\beta;i_1\ldots i_j}^{(l)} \frac{\partial u}{\partial x_{i_1}\cdots\partial x_{i_j}}\bigg|_{\boldsymbol{x}=\boldsymbol{x}_\alpha}. \tag{4.26}$$

We refer to [7] for an in-depth mathematical study of the derivative approximation (4.26).

4.4 Implementing k-Exact Reconstructions

As explained in Sect. 4.3 a k-exact mth derivative in cell $\mathscr{T}_\alpha$ can be calculated using the tensor form (see (4.21)). This is given by the sum

$$\sum_{\beta \in \mathbb{W}_\alpha} \boldsymbol{w}_{\alpha\beta}^{(m|k)} \overline{u}_\beta \,. \tag{4.27}$$

Implementing (4.27) directly requires the list of cells $\mathscr{T}_\alpha$ in the reconstruction stencil $\mathbb{W}_\alpha$ to be stored. The implementation of such an algorithm raises the following problems:

1. A parallel calculation takes advantage of the fact that an unstructured grid can easily be partitioned into subdomains each of which is handled by a different processor. Near boundary cells may have a reconstruction stencil that overlaps the opposite domain. For those cells, all data in the reconstruction stencil must be transferred from one processor to the other. The size of the overlap of reconstruction stencils varies from cell to cell, giving data packets of different size. This is a serious impairment to the efficiency of the parallel code (capability to run n times faster with n times more processors).
2. A list of neighborhoods must be created, sorted and accessed at run time. On large grids, this gives performance degradation and memory access problems.

These issues strongly motivate the search for k-exact reconstruction algorithms based upon the Von Neumann stencil (4.3). However, k-exact reconstruction requires a stencil size greater than $\binom{k+d}{k}$ cells. In dimension $d = 3$ and for second-order derivatives($k = 2$) and third-order derivatives ($k = 3$) reconstructions, this number is in general larger than the Von Neumann stencil size. The next section shows how to overcome this problem.

4.4.1 An Efficient Reconstruction Algorithm

Let $\mathbb{W}_\alpha$ be a small grid neighborhood of cell $\mathscr{T}_\alpha$, typically the Von Neumann neighborhood (4.3). The cells in $\mathbb{W}_\alpha \setminus \{\alpha\}$ are labelled with $\beta \in \{\beta_1, \ldots, \beta_m\}$, where $m \triangleq |\mathbb{W}_\alpha \setminus \{\alpha\}|$.

The suggested algorithm is to *recursively calculate* k-exact kth derivatives $\boldsymbol{w}_\beta^{(k|k)}$. This idea is well-known in the finite-difference context. Recall for example that the standard three-point second-order derivative δ_x^2 defined by $\delta_x^2 u_j = (u_{j+1} + u_{j-1} -$

$2u_j)/h^2$ is the same as $\delta_x^2 = \delta_x \circ \delta_x$ where the notation δ_x stands for the staggered difference operator $\delta_x u_j = (u_{j+1/2} - u_{j-1/2})/h$. The operator δ_x^2 is thus recursively obtained from the knowledge of δ_x.

Coming back to our problem, suppose that k-exact kth derivative operators $\boldsymbol{w}_\beta^{(k|k)}$ at cells $\mathscr{T}_\beta$ for all β are known.

Let us introduce the operator $\mathfrak{J}_{\mathbb{W}_\alpha}^{(k+1)}$,

$$\mathfrak{J}_{\mathbb{W}_\alpha}^{(k+1)} : \mathrm{S}^{k+1}\left(\mathbb{R}^d\right) \longrightarrow \left(\mathrm{S}^k\left(\mathbb{R}^d\right)\right)^m. \tag{4.28}$$

The image of the symmetric tensor $\boldsymbol{b} \in \mathrm{S}^{k+1}$ is

$$\left(\mathfrak{J}_{\mathbb{W}_\alpha}^{(k+1)}(\boldsymbol{b})\right)_\beta \in \left(\mathrm{S}^k\left(\mathbb{R}^d\right)\right)^m. \tag{4.29}$$

The βth component of (4.29) is defined by

$$\begin{aligned}\left(\mathfrak{J}_{\mathbb{W}_\alpha}^{(k+1)}(\boldsymbol{b})\right)_\beta &\triangleq \boldsymbol{h}_{\alpha\beta} \cdot \boldsymbol{b} \\ &+ \frac{1}{(k+1)!} \sum_{\gamma \in \mathbb{W}_\beta^{(k)}} \boldsymbol{w}_{\beta\gamma}^{(k|k)} \left(\boldsymbol{z}_{\beta\gamma}^{(k+1)} \bullet \boldsymbol{b}\right) - \frac{1}{(k+1)!} \sum_{\gamma \in \mathbb{W}_\alpha^{(k)}} \boldsymbol{w}_{\alpha\gamma}^{(k|k)} \left(\boldsymbol{z}_{\alpha\gamma}^{(k+1)} \bullet \boldsymbol{b}\right).\end{aligned} \tag{4.30}$$

Using index notation (4.30) can be expressed as

$$\begin{aligned}\mathfrak{J}_{\mathbb{W}_\alpha}^{(k+1)}(\boldsymbol{b})_{\beta;i_1\cdots i_k} &\triangleq \sum_{j=1}^{d} h_{\alpha\beta;j}\, b_{i_1\cdots i_k j} \\ &+ \frac{1}{(k+1)!} \sum_{\gamma \in \mathbb{W}_\beta^{(k)}} w_{\beta\gamma}^{(k|k)} \sum_{i_1\cdots i_{k+1}=1}^{d} z_{\beta\gamma;i_1\cdots i_{k+1}}^{(k+1)}\, b_{i_1\cdots i_{k+1}} \\ &- \frac{1}{(k+1)!} \sum_{\gamma \in \mathbb{W}_\alpha^{(k)}} w_{\alpha\gamma}^{(k|k)} \sum_{i_1\cdots i_{k+1}=1}^{d} z_{\alpha\gamma;i_1\cdots i_{k+1}}^{(k+1)}\, b_{i_1\cdots i_{k+1}}.\end{aligned} \tag{4.31}$$

Suppose now that $\mathbb{W}_\alpha$ is a grid neighborhood of cell $\mathscr{T}_\alpha$. Let $\boldsymbol{w}_\beta^{(k|k)}$ for $\beta \in \mathbb{W}_\alpha$ be a family of k-exact kth derivatives at cell $\mathscr{T}_\beta$ with respective reconstruction stencils $\mathbb{W}_\beta^{(k)}$ and let the function u be $(k+1)$-exact on $\mathbb{U}_\alpha \triangleq \bigcup_{\beta \in \mathbb{W}_\alpha} \mathbb{W}_\beta^{(k)}$. Then we have the following identity

$$\mathfrak{J}_{\mathbb{W}_\alpha}^{(k+1)}\left(D^{(k+1)}u\Big|_{\boldsymbol{x}_\alpha}\right) = \left(\boldsymbol{w}_{\beta_1}^{(k|k)}[\overline{\mathrm{u}}] - \boldsymbol{w}_\alpha^{(k|k)}[\overline{\mathrm{u}}], \ldots, \boldsymbol{w}_{\beta_m}^{(k|k)}[\overline{\mathrm{u}}] - \boldsymbol{w}_\alpha^{(k|k)}[\overline{\mathrm{u}}]\right). \tag{4.32}$$

The *main claim* is the following:

Claim. *Assume that the operator* $\mathfrak{J}_{\mathbb{W}_\alpha}^{(k+1)}$ *defined by (4.30) has a left inverse* $\mathfrak{D}_{\mathbb{W}_\alpha}^{(k+1)}$. *Then in that case the linear operator* $\boldsymbol{w}_\alpha^{(k+1|k+1)} : \mathbb{R}^N \longrightarrow \mathrm{S}^{k+1}\left(\mathbb{R}^d\right)$ *defined by*

$$\boldsymbol{w}_\alpha^{(k+1|k+1)}\left[\overline{\mathrm{u}}\right] \triangleq \mathfrak{D}_{\mathbb{W}_\alpha}^{(k+1)}\left(\boldsymbol{w}_{\beta_1}^{(k|k)}\left[\overline{\mathrm{u}}\right] - \boldsymbol{w}_\alpha^{(k|k)}\left[\overline{\mathrm{u}}\right], \ldots, \boldsymbol{w}_{\beta_m}^{(k|k)}\left[\overline{\mathrm{u}}\right] - \boldsymbol{w}_\alpha^{(k|k)}\left[\overline{\mathrm{u}}\right]\right) \tag{4.33}$$

is a $(k+1)$*-exact* $(k+1)$ *th derivative on the grid neighborhood* $\bigcup_{\beta\in\mathbb{W}_\alpha} \mathbb{W}_\beta^{(k)}$.

The claim holds for the following reason: let the function u be $(k+1)$-exact on $\bigcup_{\beta\in\mathbb{W}_\alpha} \mathbb{W}_\beta^{(k)}$. Therefore u and the family of $\boldsymbol{w}_\beta^{(k|k)}\left[\overline{\mathrm{u}}\right]$ satisfy the identity (4.32). Applying the operator $\mathfrak{D}_{\mathbb{W}_\alpha}^{(k+1)}$ to both sides of (4.32) gives the desired result

$$\boldsymbol{w}_\alpha^{(k+1|k+1)}\left[\overline{\mathrm{u}}\right] = D^{(k+1)}u\Big|_{\boldsymbol{x}_\alpha} . \tag{4.34}$$

The construction of the $(k+1)$-exact $(k+1)$ th derivative (4.33) thus depends on the existence of a left inverse of (4.28). Unfortunately, an abstract proof of existence of a left inverse of (4.28) is a *difficult question on arbitrary unstructured grids*. Nevertheless, we show in the next section how to take advantage of (4.33) to define a k-exact mth derivatives $\boldsymbol{w}_\alpha^{(m|k)}\left[\overline{\mathrm{u}}\right]$ for $1 \leq m \leq k-1$.

4.4.2 Practical Implementation

In this section we specify a k-exact reconstruction algorithm based on the k-exact kth derivative defined by the recurrence relation (4.33).

Since the operator $\mathfrak{J}_{\mathbb{W}_\alpha}^{(k)}$ (4.28) is in general not surjective, we need to properly define a specific inverse $\mathfrak{D}_{\mathbb{W}_\alpha}^{(k)}$. Selecting this inverse is based upon numerical efficiency. In Theorem 3.14 in [9] it is proven that the Moore Penrose inverse $J^\dagger$ of a matrix J is the left inverse of J that minimizes all *unitarily invariant* matrix norms. These norms have the property that $\|UJV\| = \|J\|$ for all unitary matrices U, V. This includes the Frobenius norm, the spectral norm, the trace norm, the Ky-Fan norms, etc. We found that selecting the Moore Penrose inverse is a suitable choice for minimizing numerical errors. In practice the Moore-Penrose inverse is evaluated using the SVD decomposition [6]. This brings us to the following

Algorithm 1 (Coupled Least Squares: calculating k-exact k-th derivative).

1. *Calculate a* 1*-exact* 1*st derivative* $\boldsymbol{w}_\beta^{(1|1)}\left[\overline{\mathrm{u}}\right]$ *from the cell averages on the Von Neumann neighborhood* $\mathbb{V}_\alpha^{(1)}$ *(4.3) of each cell* $\mathscr{T}_\alpha$ *using least squares [9].*

2. *Iterate the following step from* $m = 1$ *to* $m = k - 1$*: using (4.33), compute in each cell* $\mathscr{T}_\alpha$ *a* $(m+1)$*-exact* $(m+1)$*th derivative* $\boldsymbol{w}_\alpha^{(m+1|m+1)}[\overline{\mathrm{u}}]$ *from the* m*-exact* m*th derivatives* $\boldsymbol{w}_\beta^{(m|m)}[\overline{\mathrm{u}}]$ *for* $\beta \in \mathbb{V}_\alpha^{(1)}$.
3. *Use recursively error formula (4.25) to calculate the* k*-exact* m*th derivatives* $\boldsymbol{w}_\alpha^{(m|k)}[\overline{\mathrm{u}}]$ *for* $1 \leq m \leq k-1$.

Note that Algorithm 1 closely emulates the differentiation procedure of a smooth function where the $(m+1)$ th pointwise derivative is obtained from the mth derivative on a neighborhood of that point.

In the three-dimensional case, the Von Neumann neighborhood on tetrahedral grids consists of five cells. Numerical evidence [9] shows that 1-exact reconstruction on this stencil leads to ill-conditioned problems. It was observed that overcoming this problem can be obtained by increasing the width of the reconstruction stencil. In practice steps 1 and 2 in Algorithm 1 are modified as follows: after calculating the m-exact mth derivative $\boldsymbol{w}_\alpha^{(m|m)}[\overline{\mathrm{u}}]$ in each cell $\mathscr{T}_\alpha$, replace $\boldsymbol{w}_\alpha^{(m|m)}[\overline{\mathrm{u}}]$ with the smoothed value

$$\tilde{\boldsymbol{w}}_\alpha^{(m|m)}[\overline{\mathrm{u}}] = \frac{\sum_{\beta \in \mathbb{V}_\alpha^{(1)}} \left|\mathscr{T}_\beta\right| \boldsymbol{w}_\beta^{(m|m)}[\overline{\mathrm{u}}]}{\sum_{\gamma \in \mathbb{V}_\alpha^{(1)}} \left|\mathscr{T}_\gamma\right|}. \tag{4.35}$$

It can easily be shown that (4.35) again is a m-exact mth derivative at cell $\mathscr{T}_\alpha$ on the grid neighborhood $\tilde{\mathbb{W}}_\beta^{(m)} \triangleq \bigcup_{\beta \in \mathbb{V}_\alpha^{(1)}} \mathbb{W}_\beta^{(m)}$ where $\mathbb{W}_\beta^{(m)}$ is the reconstruction stencil of $\boldsymbol{w}_\beta^{(m|m)}[\overline{\mathrm{u}}]$.

4.5 Numerical Convergence of k-Exact Reconstructions

In this paper we present numerical results only for the grid convergence of the approximation error $\|w[\overline{\mathrm{u}}] - u\|_{\mathscr{L}_\infty(\Omega)}$.

The test case considered consists of calculating the derivatives of the two functions defined on the unit cube by

$$u(x, y, z) = \sin(2\pi x)\sin(2\pi y)\sin(2\pi z) \tag{4.36}$$

$$u(x, y, z) = 30\exp\left(-40(x^2 + y^2 + z^2)\right). \tag{4.37}$$

Periodic boundary conditions are applied in both cases. We use three kinds of grid with different shapes: a tetrahedral grid, a Cartesian grid and a grid made of general polyhedra. In the latter case, cells can have up to 20 faces.

Let $w[\overline{\mathrm{u}}]$ denote the approximant (4.6) to the function u. We assume that cell averages are given with at least k-order accuracy. In practice this is obtained by using cubature formulas exact for polynomials of degree $k = 3$. The approximation error $\|w[\overline{\mathrm{u}}] - u\|_{\mathscr{L}_\infty(\Omega)}$ is numerically evaluated as follows. Denote by $V_{\alpha\beta}$ the set of vertices spanning the face $\mathscr{A}_{\alpha\beta}$ and define for each cell $\mathscr{T}_\alpha$ the set of vertices

Table 4.1 Convergence on tetrahedral grids for the function (4.36)

$h_{\max}$	N	DLS D1(2)	DLS D2(3)	CLS D2(4)	DLS D3(4)	CLS D3(6)
0.23632	4,271					
0.18863	12,817	1.2493	4.3705	3.6689	3.6081	4.2008
0.15720	22,493	2.1638	2.3196	0.90776	3.5680	4.3848
0.12751	39,518	2.0075	2.5284	4.0729	5.3495	4.3547
0.099903	77,770	1.0628	2.3637	1.7146	2.0623	3.0495
0.079988	192,972	3.1449	2.0579	2.5916	5.9683	5.1498
		1.8966	2.7426	2.6210	4.0738	4.1966

Table 4.2 Convergence on Cartesian grids for the function (4.36)

$h_{\max}$	N	DLS D1(1)	CLS D2(2)	DLS D2(2)	CLS D3(3)	DLS D3(3)
0.10825	4,096					
0.078730	10,648	1.9930	3.0350	2.7834	4.1130	4.0078
0.061859	21,952	2.0918	3.0251	3.0007	4.0649	4.0170
0.050943	39,304	2.0121	3.0177	2.9003	4.1084	4.0440
0.043301	64,000	2.0688	3.0127	3.0146	4.0631	4.0268
0.037653	97,336	2.0135	3.0096	2.9368	4.0559	4.0482
0.033309	140,608	2.0554	3.0089	3.0174	3.9505	3.9696
0.029863	195,112	2.0125	3.0061	2.9559	4.0147	4.0865
		2.0338	3.0202	2.9246	4.0671	4.0249

Table 4.3 Convergence on polyhedral grids for the function (4.36)

$h_{\max}$	N	DLS D1(1)	CLS D2(2)	DLS D2(2)	CLS D3(3)	DLS D3(3)
0.084952	12,083					
0.080261	22,983	4.9195	12.521	10.677	6.9120	10.221
0.053768	52,487	1.2017	1.7349	2.1258	2.8336	2.4397
0.042933	80,995	2.5688	2.1967	2.0627	4.6916	3.9082
0.040684	135,609	2.9787	8.7158	6.7513	7.7399	7.3488
0.027591	238,993	1.0183	1.3725	1.9126	2.4008	2.4448
		1.6845	2.5807	2.6924	3.4964	3.3630

$$X_\alpha \triangleq \{\boldsymbol{x}_\alpha\} \cup \bigcup_{\beta \in \mathbb{V}_\alpha^{(1)}} \{\boldsymbol{x}_{\alpha\beta}\} \cup \bigcup_{\beta \in \mathbb{V}_\alpha^{(1)}} V_{\alpha\beta} \,. \tag{4.38}$$

The estimate of the approximation error is thus defined as

$$\|w\left[\overline{\mathrm{u}}\right] - u\|_{\mathscr{L}_\infty(\Omega)} \approx \sup_{1 \le \alpha \le N} \sup_{\boldsymbol{x} \in X_\alpha} \left| w_\alpha\left[\overline{\mathrm{u}}\right](\boldsymbol{x}) - u(\boldsymbol{x}) \right| \,. \tag{4.39}$$

Tables 4.1–4.3 display the convergence rate of the approximation error (4.39) as a function of the maximum cell diameter $h_{\max}$ on the three different grid shapes for the function (4.36). Tables 4.4–4.6 display the same error for the function (4.37). Each row except the last one represents a grid with the column N displaying the number of cells and the column $h_{\max}$ the maximum cell diameter. The remaining columns show for each approximation method the convergence rate of the error

Table 4.4 Convergence on tetrahedral grids for the function (4.37)

$h_{\max}$	N	DLS D1(2)	DLS D2(3)	CLS D2(4)	DLS D3(4)	CLS D3(6)
0.23632	4,271					
0.18863	12,817	1.8947	3.9364	2.2439	5.5639	5.2849
0.15720	22,493	−0.66561	2.4186	2.5729	2.2471	2.3350
0.12751	39,518	3.1658	2.0958	4.0293	2.5890	2.3286
0.099903	77,770	0.68881	3.2331	2.2069	3.6160	3.6302
0.079988	192,972	2.1892	3.4054	2.3709	4.9739	5.6719
		1.4983	3.0579	2.6617	3.8710	3.9239

Table 4.5 Convergence on Cartesian grids for the function (4.37)

$h_{\max}$	N	DLS D1(1)	CLS D2(2)	DLS D2(2)	CLS D3(3)	DLS D3(3)
0.10825	4,096					
0.078730	10,648	1.9049	3.2080	2.2145	4.1410	3.7255
0.061859	21,952	1.9113	2.8185	2.9336	3.8385	4.0430
0.050943	39,304	1.9455	3.0864	2.8520	3.9288	4.1242
0.043301	64,000	1.9628	3.0955	2.7264	3.9679	4.1314
0.037653	97,336	1.9740	2.9818	3.0550	3.9878	4.2836
0.033309	140,608	1.9802	3.2079	2.9015	3.9861	4.0752
0.029863	195,112	1.9844	2.9120	2.8733	4.0017	4.3373
		1.9410	3.0530	2.7224	3.9874	4.0422

Table 4.6 Convergence on polyhedral grids for the function (4.37)

$h_{\max}$	N	DLS D1(1)	CLS D2(2)	DLS D2(2)	CLS D3(3)	DLS D3(3)
0.084952	12,083					
0.080261	22,983	9.7162	13.505	11.200	19.321	17.962
0.053768	52,487	1.5343	2.1122	2.0169	2.8475	2.8623
0.042933	80,995	1.3044	1.6592	1.6647	2.2736	2.5098
0.040684	135,609	7.5454	10.826	10.398	15.185	15.018
0.027591	238,993	0.96482	1.4292	1.3625	1.8798	1.9781
		1.9923	2.7777	2.5850	3.8202	3.8302

when the grid is refined from the grid of the row immediately above to the grid of the current row. The last row of each table shows the convergence rate when the grid is refined from the coarsest to the finest grid.

The tests show results for piecewise polynomial approximation of degree $k = 2$ and $k = 3$ calculated using the *Coupled Least Squares* Algorithm 1. We use the nomenclature CLS D$k(n)$. The values $k = 2, 3$ denote the reconstruction order and n indicates the effective reconstruction stencil as defined by (4.4). Note that on tetrahedral grids, Algorithm 1 is modified along the smoothing formula (4.35). Direct least squares reconstruction results are displayed for comparison. The results are called DLS D$k(n)$ for $k = 1, 2, 3$ and n specifying the size of the reconstruction stencil.

The values of the convergence rate depend on the grid type and the approximated function. Theoretical considerations indicate that k-exact reconstruction should lead

to a convergence rate of $k+1$. Observe that this value is effectively obtained for the Cartesian grids. On the tetrahedral and polyhedral grids, the measured convergence rates vary much more from one row to the next than on the Cartesian grids. Note that the convergence rate from the coarsest to the finest grid that is displayed in the last row of each table is more significant because the difference in $h_{\max}$ is larger. On the unstructured grids, this rate is observed to be in $[k+\frac{1}{2}, k+1]$.

Since the main purpose of Algorithm 1 is to replace the direct approximation method DLS, the absolute values of the convergence rate seem less important to the authors than the following comparative results:

1. The CLS method defined by Algorithm 1 produces convergence rates that are close to those of the direct approximation method DLS for all three shapes of grid.
2. Increasing the degree of the polynomial reconstruction by one leads to a roughly identical increase in the convergence rate of the numerical error.

4.6 Isothermal Vortex Test-Case

In this section we consider the bidimensional Euler equations of hydrodynamics discretized by the cell centered finite volume method. The reconstruction procedure using Algorithm 1 with $k = 3$ expounded in Sect. 4.4 has been implemented combined with classical upwind fluxes such as the Roe numerical flux [11]. The test-case aims to show how the cubic reconstruction procedure compares with the classical piecewise affine reconstruction. It consists of an isothermal Gaussian vortex moving across a periodic box at constant velocity. Since we only focus on the accuracy of the spatial approximation, no slope limitation was used. The calculation was operated in parallel on 16 processors. The specific characteristics of the vortex are as follows:

$$\begin{cases} \text{gas constant: } R = 8.314 \text{ J mol}^{-1}\text{K}^{-1}, \\ \text{molar mass: } M_g = 28.013410^3 \text{ kg mol}^{-1}, \\ \sigma = 0.001, \quad a = \frac{1}{2}\left(\frac{0.005}{\sigma}\right)\frac{M_g}{RT_0} \simeq 0.000140393, \\ \text{constant temperature: } T = T_0 = 300 \text{ K}, \\ \text{Gaussian pressure distribution: } p = p_0(1 - ae^{-\frac{r^2}{2\sigma^2}}) \text{ Pa}, \\ p_0 = 10^5 \text{ Pa}, \text{(gives } p_{min} \simeq 99985.96 \text{ Pa)}, \\ \text{vortex angular velocity: } v_\theta = \sqrt{\frac{2aRT_0/M_g}{1-ae^{-\frac{r^2}{2\sigma^2}}}} e^{-\frac{r^2}{2\sigma^2}}, \\ \text{advection velocity: } v_x = 50 \text{ ms}^{-1}, \quad v_y = 20 \text{ ms}^{-1}, \\ t_{start} = 0, \quad t_{end} = 0.004, \\ 10 \text{ periods along the } x \text{ direction}, \\ 4 \text{ periods along the } y \text{ direction}, \\ \Delta t = 5.10^{-8} \text{ s}, \quad \text{CFL on finest grid} \simeq 0.2. \end{cases} \tag{4.40}$$

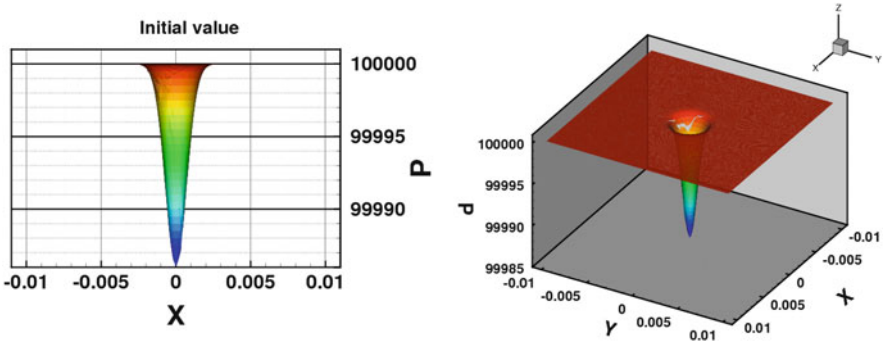


Fig. 4.2 Convection of the isothermal vortex (4.40) – initial condition

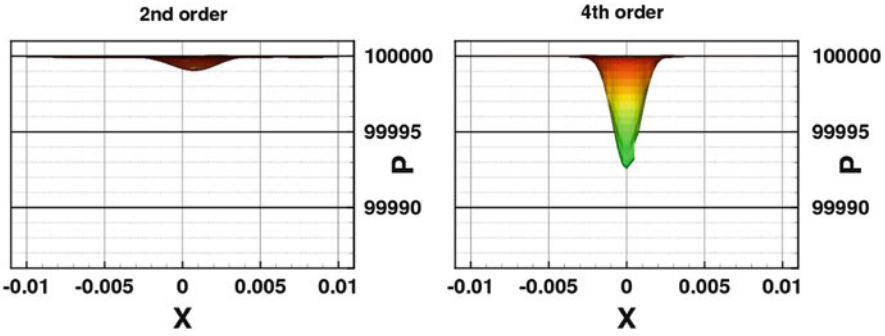


Fig. 4.3 The isothermal vortex (4.40) – Cartesian coarse grid – final time

This vortex is an analytical solution of the Euler equations (Fig. 4.2). Different grids were tested:

- Coarse Cartesian grid (80×80, span of the vortex: 8 cells), fine Cartesian grid (200×200, span of the vortex: 20 cells)
- Coarse unstructured grid (span of the vortex: $\simeq$10 cells), fine unstructured grid (span of the vortex: $\simeq$15 cells).

The reconstruction procedure follows the lines of Algorithm 1 with cubic reconstruction ($k = 3$). The RK4 time-stepping scheme is used. We report on Figs. 4.3–4.6 the numerical damping between the classical second-order (piecewise affine reconstruction) and the new fourth-order (cubic reconstruction) finite-volume schemes. The results clearly show the drastic influence offered by the cubic reconstruction even with the optional smoothing step (4.35), see Fig. 4.6.

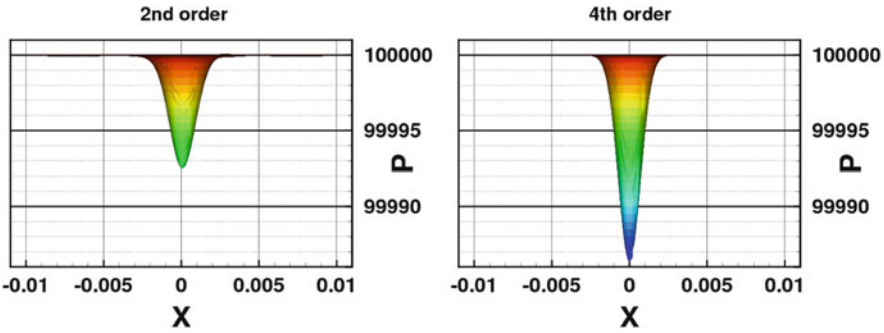


Fig. 4.4 Convection of the isothermal vortex (4.40) – Cartesian fine grid – final time

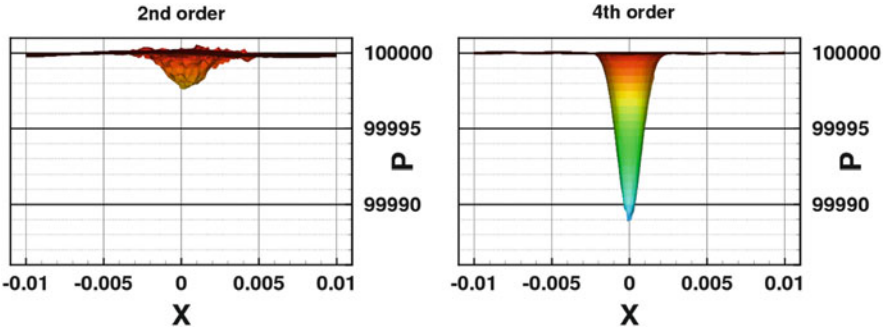


Fig. 4.5 Convection of the isothermal vortex (4.40) – unstructured coarse grid – final time

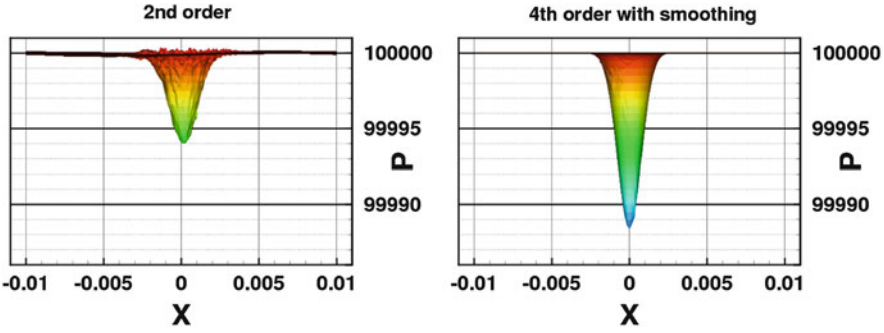


Fig. 4.6 Convection of the isothermal vortex (4.40) – unstructured fine grid – final time

4.7 Conclusion

This paper presents the mathematical foundation and the numerical implementation of a new algorithm for the calculation of k-exact polynomial approximants. This algorithm can be used for coding $(k+1)$-order accurate finite volume schemes as

introduced in [4]. It is well suited for parallel architectures since the approximation procedure in a grid cell uses only data from adjacent cells. By operating in several steps, the algorithm accesses data from grid cells beyond the adjacent cells in an implicit and natural way. The paper contains a thorough numerical validation of the new algorithm on Cartesian and unstructured grids showing promising results regarding applications for complex fluid flows. More mathematical details of the procedure can be found in the forthcoming [7].

Acknowledgements The authors acknowledge P.L. Roe for his help in preparing the final version of this paper.

References

1. Abgrall, R.: On essentially non-oscillatory schemes on unstructured meshes: analysis and implementation. J. Comput. Phys. **114**, 45–58 (1994)
2. Abgrall, R., Sonar, T.: On the use of Mühlbach expansions in the recovery step of ENO methods. Numer. Math. **76**, 1–25 (1997)
3. Barth, T.J.: Recent developments in high order k-exact reconstruction on unstructured meshes. In: AIAA 93, AIAA-93-0668, pp. 1–15. AIAA, Reno (1993)
4. Barth, T.J., Frederickson, P.O.: Higher order solution of the Euler equation on unstructured grids using quadratic reconstruction. In: AIAA 90, AIAA-90-0013, pp. 1–12. AIAA, Reno Nevada (1990)
5. Delanaye, M., Essers, J.A.: Quadratic-reconstruction finite volume scheme for compressible flows on unstructured adaptive grids. AIAA J. **35**(4), 631–639 (1997)
6. Golub, G.H., Van Loan, C.F.: Matrix Computations, 3rd edn. John Hopkins University Press, Baltimore (1996)
7. Haider, F.: High order approximation on unstructured grids: theory and implementation (preprint, 2013)
8. Haider, F., Brenner, P., Courbet, B., Croisille, J.P.: Efficient implementation of high order reconstruction in finite volume methods. In: Dell'Acqua, F.C., Reizakis, M., Herbin, R., Halama, J., Fürst, J. (eds.) Sixth International Symposium on Finite Volumes for Complex Applications, Prague, pp. 553–562. Springer (2011)
9. Haider, F., Croisille, J.-P., Courbet, B.: Stability analysis of the cell centered finite-volume MUSCL method on unstructured grids. Numer. Math. **113**, 555–600 (2009)
10. Iske, A., Sonar, T.: On the structure of function spaces in optimal recovery of point data for ENO-schemes by radial basis functions. Numer. Math. **74**, 177–201 (1996)
11. Roe, P.L.: Approximate Riemann solvers, parameter vectors and difference schemes. J. Comput. Phys. **43**, 357–372 (1981)
12. Shu, C.-W.: High order ENO and WENO schemes. In: Barth, T.J., Deconinck, H. (eds.) High-Order Methods for Computational Physics. Lecture Notes in Computer Science and Engineering, pp. 439–582. Springer, Berlin/New York (1999)

Chapter 5
3D Application of Higher Order Multigrid Algorithms for a RANS-$k\omega$ DG-Solver

Marcel Wallraff and Tobias Leicht

Abstract We present an application of multigrid algorithms to a Discontinuous Galerkin discretization of the 3D Reynolds-averaged Navier–Stokes equations in combination with a $k\omega$ turbulence model. Based on either lower order discretizations or agglomerated coarse meshes the resulting algorithms can be characterized as p- or h-multigrid, respectively. Linear and nonlinear multigrid algorithms are applied to a 3D numerical test case, namely the VFE-2 delta-wing with rounded leading edge. All presented algorithms are compared to a strongly implicit single grid solver in terms of run time behavior and nonlinear iterations.

5.1 Introduction

The Reynolds-averaged Navier–Stokes (RANS) equations in combination with suitable turbulence models such as the $k\omega$ model of Wilcox [19] can still be considered as state-of-the-art for many applications in exterior aerodynamics. It has been shown that Discontinuous Galerkin (DG) finite element methods are suitable for CFD application in the area of compressible aerodynamic flows [2, 6, 9, 11]. One of the main drawbacks of DG methods is the relatively high computation cost per degree of freedom. High Reynolds numbers, the associated highly stretched meshes typically used for an optimal resolution of turbulent boundary layers, and source terms present in the turbulence models all contribute to an increased stiffness of the resulting algebraic system of equations that has to be solved. Implicit operators are generally accepted as key component for increased efficiency of iterative algorithms. Several authors suggested strongly implicit schemes that are

M. Wallraff (✉) • T. Leicht
German Aerospace Center (DLR), Institute of Aerodynamics and Flow Technology,
Lilienthalplatz 7, 38108 Braunschweig, Germany
e-mail: marcel.wallraff@dlr.de

R. Abgrall et al. (eds.), *High Order Nonlinear Numerical Schemes for Evolutionary PDEs*, Lecture Notes in Computational Science and Engineering 99,
DOI 10.1007/978-3-319-05455-1_5,

close to Newton's method [2, 9]. It is well-known that these implicit methods work best at the end of the iterative procedure in the regime of asymptotic convergence. Employing a mesh or order sequencing as start-up strategy helps to alleviate this problem by providing a good initial guess, see Sect. 5.3 for further information. In this work a 3D application of nonlinear h- and p-multigrid algorithms are investigated as well as linear h- and p-multigrid algorithms used as a preconditioner.

Previous studies of these algorithms in 2D included a three element airfoil (L1T2) test case [7] and test case C3.1 (MDA 30P-30N multi-element airfoil) of the 2nd International Workshop on High-Order CFD Methods. These investigations of the presented algorithms in 2D are a baseline for the evaluation of the first 3D application. The 2D investigations are not published yet. Thus, in the following a short summary is given. 2D results indicate that the best performance in algorithmic convergence and run-time behavior is achieved with a combination of a nonlinear multigrid with a Backward-Euler smoother on each level, in which the resulting linear systems are solved with a linear multigrid preconditioned GMRes method. These findings indicate also that the nonlinear multigrid algorithms are beneficial at the beginning of the computation on every level and the linear multigrid algorithms are superior at the end of the iterative procedure in the regime of asymptotic convergence. Thus, the combination of a nonlinear and linear multigrid accomplishes the biggest reduction in nonlinear iterations for both the p- and h-start-up computations. Moreover, an h-start-up strategy is in most cases much more favorable than a p-start-up for finer meshes. Furthermore, the h-multigrid algorithms seems to be both more robust and faster in terms of run-time behavior, in particular for finer meshes.

We will present 3D numerical examples in order to assess whether the performance of the proposed algorithms with respect to both algorithmic convergence properties and run-time behavior is similar to previous findings in 2D, as summarized briefly above. The presentation of 3D results concentrates on the VFE-2 delta-wing with rounded leading edge, see Sect. 5.3.

5.2 Discretization

A DG discretization of the steady-state RANS-$k\omega$ equations [19, 20] are briefly introduced. These equations can be formulated in conservative form based on convective fluxes $\mathscr{F}^c$ and viscous fluxes $\mathscr{F}^v$, corresponding to first and second order derivatives, respectively, and additional source terms $\mathscr{S}(\mathbf{u}, \nabla\mathbf{u})$. The $k\omega$ turbulence model consists of conservation equations for the turbulent kinetic energy k and the turbulence dissipation rate ω. We consider the flux formulation of the steady-state RANS-$k\omega$ equations [19, 20]

$$\nabla \cdot (\mathscr{F}^c(\mathbf{u}) - \mathscr{F}^v(\mathbf{u}, \nabla\mathbf{u})) = \mathscr{S}(\mathbf{u}, \nabla\mathbf{u}), \tag{5.1}$$

with the vector of conservative variables $\mathbf{u} = (\rho, \rho v_1, \ldots, \rho v_d, \rho E, \rho k, \rho\omega)^T$, where ρ denotes the density, $\mathbf{v} = (v_1, \ldots, v_d)^T$ the velocity vector, E the total energy, k the turbulent kinetic energy and ω the turbulence dissipation rate. This model sets the eddy viscosity μ_t to

$$\mu_t := \frac{\rho k}{\omega} \tag{5.2}$$

and includes two transport equations for the turbulence kinetic energy k and the specific dissipation rate ω. As suggested by Bassi et al. [2] the transformation $\tilde{\omega} = \ln(\omega)$ is applied. Furthermore, a limitation for $\tilde{\omega}$ based on realizability constraints is introduced. Note, that the limitation on $\tilde{\omega}$ avoids unphysical values and has been found in [2, 9, 15] to have an stabilizing effect on the numerical scheme.

Furthermore, the DG discretization of these equations follows standard techniques based on numerical fluxes. For the viscous terms, the numerical representation of $\mathscr{F}^v(\mathbf{u}, \nabla\mathbf{u})$, the second scheme of Bassi and Rebay (BR2) [1, 2] is applied. The numerical representation of $\mathscr{F}^c(\mathbf{u})$, namely the convective fluxes, is given by the Roe flux [16]. All equations are treated fully coupled, i.e., there is a single Jacobian used in the eigenvalue decomposition [18]. An entropy fix is used in order to ensure non-vanishing dissipation.

An ortho-normal non-parametric formulation of the basis functions directly in the physical space will be used. This space $\mathbf{V}_l$ is equivalent to the choice of a Taylor basis [12]. This way, the local basis can be defined independently from a specific reference element, which is necessary for the usage of agglomerated meshes in an h-multigrid. The choice of this space in connection with the formulation of a DG method on agglomerated meshes has been described in detail in [4, 5]. By choosing an ortho-normal basis for the DG discretization the mass-matrix is the identity matrix on each mesh element and globally over the whole mesh.

5.3 Multigrid Algorithms

We will present several strategies to exploit hierarchies of coarse level problems in solver algorithms, including both level sequencing and linear as well as nonlinear multigrid variations [17]. Based on either lower order discretizations or agglomerated coarse meshes the resulting algorithms can be characterized as p- or h-multigrid, respectively. The only difference between these multigrid algorithms is the use of different coarse level DG discretizations and, therefore, transfer operators. All other ingredients like smoothers, timestep control, usage of a Galerkin-transfer [17], start-up strategy, etc. are the same for both kinds of multigrid algorithms. The restriction transfer operator for both h- and p-multigrid algorithms of the nonlinear state vector is given by an orthogonal L^2-projection $\hat{I}_l^{l-1} : \mathbf{V}_l \to \mathbf{V}_{l-1}$ [3].

Algorithm 1 Nonlinear multigrid algorithm $NMG(\mathbf{f}_l, \mathbf{u}_l, m_1, m_2, l, l_{min}, \tau)$

if $l = l_{min}$ **then**
  Solve $\mathbf{L}_{l_{min}}(\mathbf{u}_{l_{min}}) = \mathbf{f}_{l_{min}}$
  Set Solution to $\tilde{\mathbf{u}}_{l_{min}}$.
if $l > l_{min}$ **then**
  for $i = 1$ to m_1 **do**
    $\mathbf{u}_l := GL_l^s(\mathbf{u}_l, \mathbf{f}_l)$ /* pre-smoothing */
  $\mathbf{u}_{l-1}^0 := \hat{I}_l^{l-1}\mathbf{u}_l$
  $\mathbf{f}_{l-1} := \mathbf{f}_{l-1} + I_l^{l-1}\left(\mathbf{f}_l - \mathbf{L}_l(\mathbf{u}_l)\right) - \left(\mathbf{f}_{l-1} - \mathbf{L}_{l-1}(\mathbf{u}_{l-1}^0)\right)$
  for $k = 1$ to τ **do**
    $\mathbf{u}_{l-1}^k := NMG(\mathbf{f}_{l-1}, \mathbf{u}_{l-1}^{k-1}, m_1, m_2, l-1, l_{min}, \tau)$
  $\tilde{\mathbf{u}}_l := \mathbf{u}_l + I_{l-1}^l\left(\mathbf{u}_{l-1}^\tau - \mathbf{u}_{l-1}^0\right)$
  for $i = 1$ to m_2 **do**
    $\tilde{\mathbf{u}}_l := GL_l^s(\tilde{\mathbf{u}}_l, \mathbf{f}_l)$ /* post-smoothing */
return $\tilde{\mathbf{u}}_l$

Algorithm 2 Backward–Euler $BWE(\mathbf{u}_{l,i-1})$

Solve $\left[(\alpha_i \Delta t)^{-1}\underline{\mathrm{M}} + \underline{\mathrm{R}}_l\right](\mathbf{u}_{l,i} - \mathbf{u}_{l,i-1}) = \left[\mathbf{f}_l - \mathbf{L}_l(\mathbf{u}_{l,i-1})\right]$,
return $\mathbf{u}_{l,i}$

Furthermore, the prolongation transfer operator is obtained via a natural injection $I_{l-1}^l : \mathbf{V}_{l-1} \rightarrow \mathbf{V}_l$ and a restriction operator for the defect between the various approximation levels is defined by $I_l^{l-1} := (I_{l-1}^l)^T$. The Galerkin-transfer is defined as $\underline{\mathrm{R}}_{l-1} = I_l^{l-1}\underline{\mathrm{R}}_l I_{l-1}^l$ for a given matrix $\underline{\mathrm{R}}_l$ which is a linear map in $\mathbf{V}_l$. Moreover the resulting matrix $\underline{\mathrm{R}}_{l-1}$ is a linear map in $\mathbf{V}_{l-1}$. The actual form of the block matrices defining the transfer operators depend on the choise of basis functions. They take a particularly simple form if ortho-normal hierarchic basis functions are used on each level. For instance, for a p-multigrid the prolongation operator has the form of an identity matrix filled with additional zero rows at the bottom.

The nonlinear multigrid (Algorithm 1) is defined recursively. For $\tau = 1$ a V-cycle is done, whereas $\tau = 2$ produces a W-cycle. The nonlinear problem on every level is defined as $\mathbf{L}_l(\mathbf{u}_l) = \mathbf{f}_l$. For the nonlinear smoother $GL_l^s(\mathbf{u}_l, \mathbf{f}_l)$ we employ as underlying relaxation scheme a linearized Backward-Euler approach based on local pseudo-time steps that can be considered as a stabilized Newton's method and is also used predominantly as single-level solver [2, 9]. The algorithm is defined as follows, where $\underline{\mathrm{R}}_l := \frac{\partial \mathbf{L}_l(\mathbf{u}_l)}{\partial \mathbf{u}_l}$ is the fully implicit Jacobian matrix and $\underline{\mathrm{M}}$ is the mass matrix. In addition to that $\mathbf{u}_{l,j}$ is a given state vector, with $\mathbf{u}_{l,j} \in \mathbf{V}_l \ \forall \ j \in \mathbb{N}$. This method is controlled by a pseudo-time stepping scheme $(\alpha_i \Delta t)$, based on local time steps computed from the local state and a given CFL number. The pseudo-time step acts as a stabilizing mechanism and for CFL $\rightarrow \infty$ the Newton algorithm is recovered. Smaller time steps and accordingly CFL numbers are required in the initial phase of the solution process when the current iterate solution approximation is still too far from the converged solution. A *switched evolution relaxation* (SER) [14] technique is employed to modify the CFL number during the solution process,

Algorithm 3 Line-Jacobi scheme $LJS(\mathrm{u}_{l,k,i-1})$

1: $\delta \mathrm{u}_{l,k,i} := \mathrm{u}_{l,k,i} - \mathrm{u}_{l,k,i-1} = \mathrm{R}_{l,k}^{-1}(\mathfrak{f}_{l,k} - \mathscr{L}_{l,k}\mathrm{u}_{l,k,i-1})$
2: $\mathrm{u}_{l,k,i} := \mathrm{u}_{l,k,i-1} + \delta \mathrm{u}_{l,k,i}$,
3: **return** $\mathrm{u}_{l,k,i}$

enabling to recover the optimal behavior of the Newton algorithm in the final phase of the solution process.

The resulting linear system is solved with a Krylov method, due to the non-symmetric Jacobian GMRes will be used. This method is preconditioned either by a line-Jacobi [18] iterative scheme or a linear multigrid using the line-Jacobi scheme as a smoother. This line-Jacobi scheme is stated as follows, let $\mathscr{L}_{l,k}(\mathrm{u}_{l,k}) = \mathfrak{f}_{l,k}$ the underlying linear problem on line k, where $\mathrm{R}_{l,k}^{-1}$ is the inverse of the Jacobian matrix computed one line k in the mesh. We consider lines of mesh elements, i.e., disjoint subsets of elements. In order to find suitable lines a scalar advection-diffusion problem is defined and examined [8]. The inversion on a line based system matrix which is a tridiagonal system can be applied exactly. However, the underlying matrix to be inverted is only an approximation of the fully implicit matrix.

The following choice of parameters is used in the numerical examples in Sect. 5.4. For a nonlinear multigrid V-cycles are used with one pre- and post-smoothing step. On the lowest level one smoother step is done as well. The smoother is a Backward-Euler with SER timestep control [14]. The linear problems resulting from the Backward-Euler method are solved with a GMRes scheme. A linear multigrid as preconditioner with four post-smoothing steps and no pre-smoothing steps is applied. In the linear multigrid a line-Jacobi scheme is used as a smoother. If no linear multigrid is applied then four line-Jacobi iterations are done as a preconditioner. For all computations a start-up strategy is applied. The basic idea is to start the computation on the lowest level and transfer the resulting solution to the next higher level to provide a suitable initial solution. On every sublevel the start-up strategy produces converged solutions according to a supplied residual reduction tolerance. A p-start-up strategy generates solutions on the same computational mesh for all discretizations from p_{min} to p_{max}. Here $p_{min} = 1$ and $p_{max} = 2$ are employed for the numerical examples. We do not consider $p = 0$ discretizations, as this does not yield a consistent first order discretization in the presence of viscous fluxes and source terms. The first order discretization could be modified, but this is not done here. Thus, second order ($p = 1$) is the lowest level considered in this work. For an h-start-up strategy the initial fine mesh is coarsened in a way such that the resulting coarse meshes build a nested hierarchy of meshes. The h-start-up strategy computes a solution on the coarsest mesh and uses this as an initial solution for the next mesh in the hierarchy until the algorithm produces a solution for the initial fine mesh. In every h-start-up strategy for this test case a p-start-up strategy is done on the coarsest mesh.

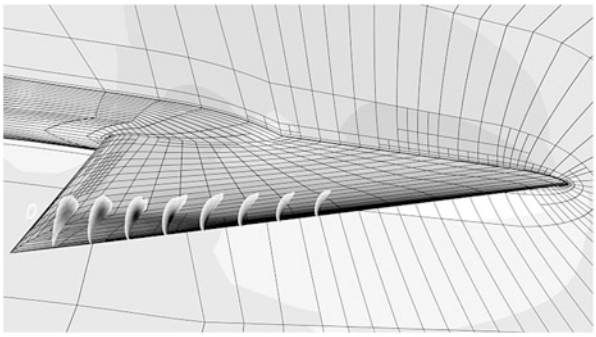


Fig. 5.1 Surface pressure plot of a $p = 2$ solution on the adjoint-based refined mesh with 23,877 elements

5.4 Numerical Results

The results in this paper represent the first application of the algorithms mentioned in Sect. 5.3 in 3D. They do not constitute an exhaustive investigation. Instead, they are used to identify areas of future research for the extension of the existing algorithms for 2D to 3D.

The VFE-2 delta-wing is a test case also considered in the EC funded IDIHOM project [10]. The flow conditions are: Mach number $= 0.4$, Reynolds number $= 3 \cdot 10^6$, $\alpha = 13.3°$ and $\beta = 0°$. Computations will be shown on a structured mesh with 13,816 elements and on an adjoint-based refined mesh with 23,877 elements. A third order accurate solution ($p = 2$) is desired for all computations (Fig. 5.1).

Only two level multigrid algorithms are shown for the current results. This is due to the coarseness of the meshes. If they are agglomerated more than once, the resulting coarse level cannot represent the domain properly.

Figure 5.2 shows the nonlinear Backward-Euler iterations on the highest level of the start-up strategies. The dashed lines in Fig. 5.2 are initialized with a $p = 1$ solution on the same mesh to compute a $p = 2$ solution. All dashed line computations use a p-start-up strategy with the same lower level computation for $p = 1$. For all solid lines the h-start-up strategy is the same as well. Since we use two level multigrid algorithms the initial solution for the solid lines in Fig. 5.2 is indeed the same. This results from the fact that on the agglomerated grid a p-start-up strategy is applied and after convergence of the $p = 2$ solution on the agglomerated grid it is used as an initial solution for the $p = 2$ computations on the mesh with 13,816 elements.

For comparability the timestep control for all computations shown in Fig. 5.2 is the same, including the same initial CFL number on the top level of the start-up strategies. The maximum number of GMRes iterations is 15 for all computations on every level. For the solid and dashed unmarked lines a single grid solver is used, namely a Backward-Euler with a line-Jacobi scheme as a preconditioner for the underlying GMRes method. Thus, the only difference between the solid and dashed

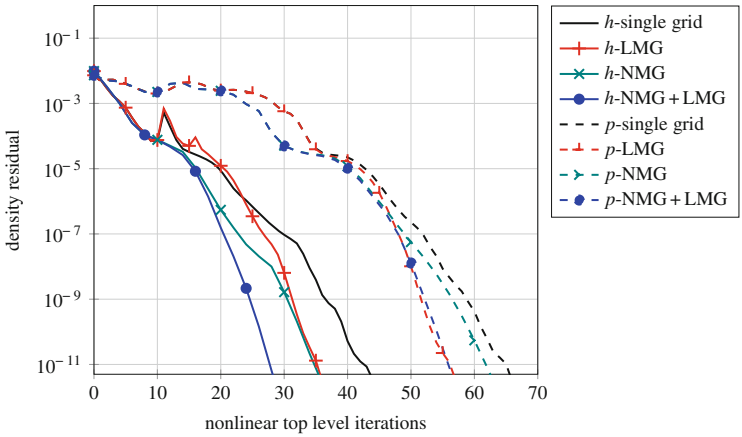


Fig. 5.2 Computation of a $p = 2$ solution on the mesh with 13,816 elements for the top level using different MG algorithms

unmarked lines is the initial solution given on this level. In general our findings are that an h-start-up is more favorable than a p-start-up, as seen in this case for a single level solver.

The lines marked with a $+$ represent a Backward-Euler solver with a linear multigrid as preconditioner for the GMRes method. The lines marked with a $\times$ indicate the use of a nonlinear multigrid with a Backward-Euler as smoother and a line-Jacobi preconditioned GMRes method. Finally, the lines marked with a $\bullet$ represent a nonlinear multigrid with Backward-Euler as smoother solved via a linear multigrid preconditioned GMRes method. Please note, that for the lines marked with $\times$, $\bullet$ and $+$ additional work which is not plotted in Fig. 5.2 is done on the coarse level of the multigrid algorithms.

The results in Fig. 5.2 are in line with previous findings from 2D cases. These findings indicate that the nonlinear multigrid algorithms are beneficial at the beginning of the computation on every level, whereas the linear multigrid algorithms are superior at the end of the iterative procedure in the regime of asymptotic convergence, where they yield an improved rate of convergence. Therefore, the combination of a nonlinear and linear multigrid accomplishes the biggest reduction in nonlinear iterations for both the p- and h-start-up computations. In addition to that the nonlinear p-multigrid alone shows a pale performance in comparison to a p-single grid computation. This supports our findings from [18].

Figures 5.3 and 5.4, which show the timings of the same computation, indicate that the better algorithmic convergence carries over to an improved runtime behavior. For the start-up strategies the coarse level convergence criterion is set to a reduction by 10^{-6}, as seen in Figs. 5.3 and 5.4. In Fig. 5.5 for the lines marked with an $\diamond$ the convergence tolerance of the start-up strategies is set to a residual reduction of 10^{-12}. The convergence plots on the top level are unaffected by this variation for all presented algorithm combinations. This is visible for the combination of a

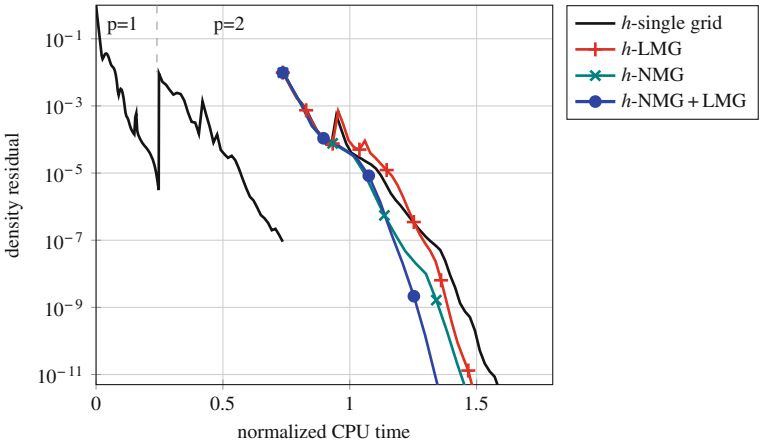


Fig. 5.3 Computation of a $p = 2$ solution for the VFE-2 delta-wing with an h-start-up strategy

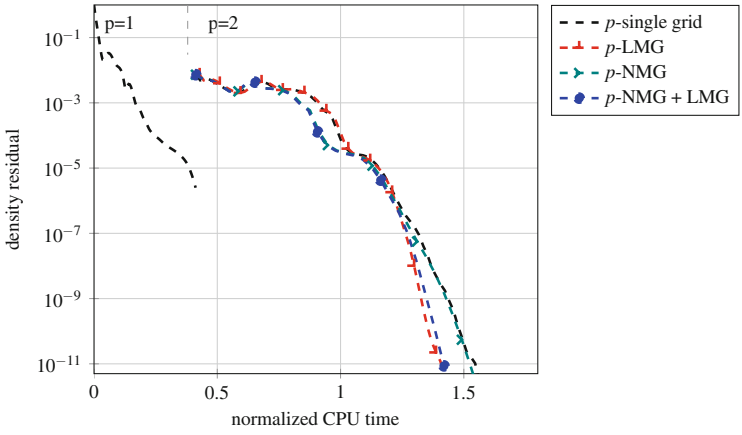


Fig. 5.4 Computation of a $p = 2$ solution for the VFE-2 delta-wing with a p-start-up strategy

nonlinear and linear multigrid algorithm from the fact that the lines marked with a $\diamond$ and $\bullet$ lie exactly on top of each other in Fig. 5.5.

Never the less the h-start-up computations in this case are not superior to their p counterparts in terms of normalized CPU time. In Fig. 5.3 most of the time is spent on the agglomerated mesh and thus on a coarse level. In contrast to that, in Fig. 5.4 most of the time is used for the $p = 2$ computation on the mesh with 13,816 elements and thus on the fine level. Although on the mesh with 13,816 elements for a $p = 2$ computation for the whole computations the h-multigrid algorithms are slightly faster than their p counterparts with the same solver settings even with a inferior start-up strategy for this coarse mesh. This results from the fact that fewer iterations are used on the highest level, as seen in Fig. 5.2. This again confirms the

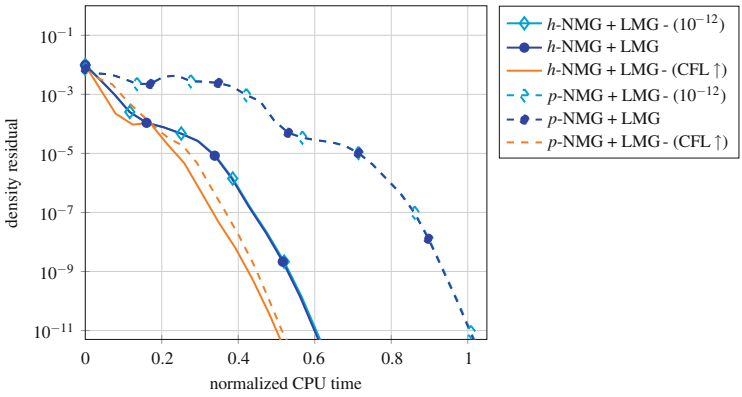


Fig. 5.5 Computation of a $p = 2$ solution on the mesh with 13,816 elements for the top level with higher initial CFL numbers for unmarked lines

findings from 2D where an h-start-up strategy is more favorable than a p-start-up strategy in most cases, see Sect. 5.1.

Moreover, a beneficial run-time behavior of the multigrid algorithms can be seen in comparison with a single grid solver, (unmarked lines in Figs. 5.2–5.4), even though additional work is done between the top-level Backward-Euler smoother iterations in the multigrid algorithms.

As mentioned in Sect. 5.1, the combination of nonlinear and linear multigrid algorithms is more robust than the other algorithms shown in Fig. 5.2 for 2D test cases. This investigation of robustness in the 3D test case can be seen in Fig. 5.5. The lines marked with a • are the same as shown before in Fig. 5.2. For the unmarked lines the same algorithms are used but for the solid line a three times higher initial CFL number and for the dashed line a five times higher initial CFL number is used. The single grid algorithms from Fig. 5.2 do not converge for these increased initial CFL numbers and the combination of both multigrid algorithms is still the best algorithm in terms of CPU time and nonlinear top level iterations. This indicates that the findings in terms of robustness also hold in 3D. Moreover, with the reduced number of nonlinear top level iterations there is also an improvement in run-time behavior. The reduction in nonlinear top level iterations in the h-multigrid is from 30 to 24 which results in an reduction of normalized CPU time for the whole computation of only 0.12. In contrast to that, in the p-multigrid case there is an enhancement of 0.46 in normalized CPU time for the whole computation. This results from the fact that the dashed unmarked line in Fig. 5.5 reduces the nonlinear top level iterations from 58 to 32 in comparison to the dashed line marked with a •. This will set the normalized CPU time of the complete computation for the unmarked dashed line in Fig. 5.5 including the start-up strategy to 1.00. Please note that for the unmarked lines in Fig. 5.5 the asymptotic convergence rate at the end of the computations is still the same than for the lines marked with a •. The higher initial CFL number seems to be beneficial at the beginning of the computation on

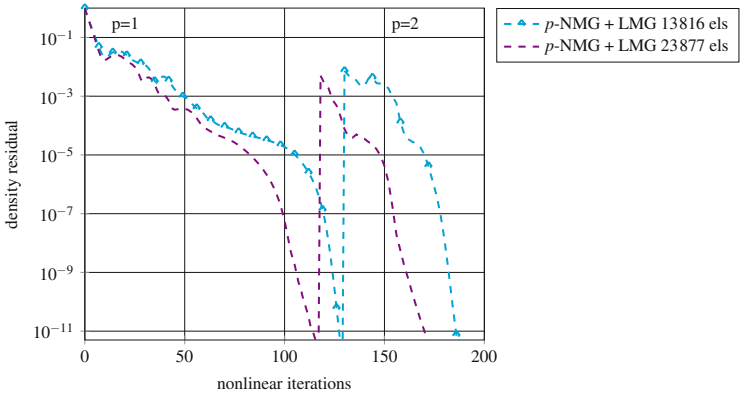


Fig. 5.6 Convergence computations on the mesh with 13,816 elements (lines with ◇) and on an adjoint-based refined mesh with 23,877 elements (unmarked lines)

this multigrid level. Due to the SER timestep control [14] which increases the CFL number at the end of the computation, with CFL$\rightarrow \infty$, the asymptotic convergence rate remains unchanged.

In Fig. 5.6 exactly the same parameters on every level were used for both computations, the only difference is the underlying mesh. The adjoint-based refined grid seems to better resolve critical areas of the flowfield and therefore the convergence behavior is slightly better than on the mesh with 13,816 elements. This constitutes an encouraging result for further investigations.

5.5 Summary and Outlook

This paper is a first presentation of a 3D application of the proposed multigrid algorithms. First results look promising, but further analysis is required due to the limited scope of the presented investigations and the consideration of rather coarse meshes.

The conclusions drawn from these computations indicate that the nonlinear multigrid algorithms are beneficial at the beginning of the computation on every level, whereas the linear multigrid algorithms are superior at the end of the iterative procedure in the regime of asymptotic convergence, where they yield an improved rate of convergence. It seems that a combination of nonlinear and linear multigrid algorithm are the fastest in terms of normalized CPU time. Moreover the combination of nonlinear and linear multigrid algorithms is more robust than the other algorithms shown in this work. We also conclude that the beneficial behavior of the presented algorithms carries over from 2D to 3D.

In future work it will be investigated if a line-Jacobi scheme in combination with a linear multigrid in 3D can perform as good as in 2D. This is of interest as the

line-Jacobi scheme for a 2D quadrilateral takes into account half of the Jacobian off diagonal blocks whereas only a third of the Jacobian off diagonal blocks is taken into account for a 3D hexahedron. The current results seem to indicate that lines can be used for preconditioning in 3D, even though the number of discarded couplings w.r.t. the full Jacobian is higher than in 2D. Moreover, the solver settings are the same as in the 2D test cases. Maybe after further investigation of 3D test cases some modifications will be made. Another issue will be the agglomeration of 3D meshes in combination with the presented algorithms. For the numerical examples, the agglomeration is based on the `MGridGen` software, which employs graph-based algorithms to optimize the ratio of agglomerate surface area to agglomerate volume [13], i.e., it aims at isotropic coarse meshes. Exploiting the available structure of meshes, in particular in boundary layers, might be beneficial.

Acknowledgements The authors acknowledge the financial support of both the BMBF joint research project DGHPOPT (Förderkennzeichen: 05M10CLA) and the EC funded IDIHOM project [10].

References

1. Bassi, F., Rebay, S., Mariotti, G., Pedinotti, S., Savini, M.: A high-order accurate discontinuous finite element method for inviscid and viscous turbomachinery flows. In: Decuypere, R., Dibelius, G. (eds.) In: 2nd European Conference on Turbomachinery Fluid Dynamics and Thermodynamics, Antwerpen, 5–7 March 1997, pp. 99–108. Technologisch Instituut (1997)
2. Bassi, F., Crivellini, A., Rebay, S., Savini, M.: Discontinuous Galerkin solution of the Reynolds-averaged Navier-Stokes and k-ω turbulence model equations. Comput. Fluids **34**(4–5), 507–540 (2005)
3. Bassi, F., Ghidoni, A., Rebay, S., Tesini, P.: High-order accurate p-multigrid discontinuous Galerkin solution of the Euler equations. Int. J. Numer. Methods Fluids **60**, 847–865 (2009)
4. Bassi, F., Botti, L., Colombo, A., Di Pietro, D.A., Tesini, P.: On the flexibility of agglomeration based physical space discontinuous Galerkin discretizations. J. Comput. Phys. **231**(1), 45–65 (2012)
5. Bassi, F., Botti, L., Colombo, A., Rebay, S.: Agglomeration based discontinuous Galerkin discretization of the Euler and Navier-Stokes equations. Comput. Fluids **61**, 77–85 (2012)
6. Burgess, N.K., Nastase, C.R., Mavriplis, D.J., Martinelli, L.: Efficient solution techniques for discontinuous Galerkin discretizations of the Navier-Stokes equations on hybrid anisotropic meshes. In: 48th AIAA Aerospace Sciences Meeting, AIAA Paper 2010-1448, Orlando (2010)
7. Fejtek, I.: Summary of code validation results for a multiple element airfoil test case. In: 28th AIAA Fluid Dynamics Conference, AIAA Paper 97-1932, Snowmass Village (1997)
8. Fidkowski, K.J., Oliver, T.A., Lu, J., Darmofal, D.L.: p-multigrid solution of high-order discontinuous Galerkin discretizations of the compressible Navier-Stokes equations. J. Comput. Phys. **207**(1), 92–113 (2005)
9. Hartmann, R., Held, J., Leicht, T.: Adjoint-based error estimation and adaptive mesh refinement for the RANS and k-ω turbulence model equations. J. Comput. Phys. **230**(11), 4268–4284 (2011)
10. Kroll, N.: IDIHOM – European project on industrialization of high-order methods for aeronautical applications. In: ECCOMAS 2012, Wien, Sept 2012
11. Landmann, B., Kessler, M., Wagner, S., Krämer, E.: A parallel, high-order discontinuous Galerkin code for laminar and turbulent flows. Comput. Fluids **37**(4), 427–438 (2008)

12. Luo, H., Baum, J.D., Löhner, R.: A discontinuous Galerkin method based on a Taylor basis for the compressible flows on arbitrary grids. J. Comput. Phys. **227**(20), 8875–8893 (2008)
13. Moulitsas, I., Karypis, G.: Multilevel algorithms for generating coarse grids for multigrid methods. In: Proceedings of the 2001 ACM/IEEE Conference on Supercomputing, Denver, p. 45. ACM (2001)
14. Mulder, W.A., Van Leer, B.: Experiments with implicit upwind methods for the Euler equations. J. Comput. Phys. **59**(2), 232–246 (1985)
15. Pelletier, D., Ilinca, F.: Positivity preservation and adaptive solution for the $k - \omega$ model of turbulence. AIAA J. **36**, 44–50 (1998)
16. Roe, P.L.: Approximate Riemann solvers, parameter vectors, and difference schemes. J. Comput. Phys. **43**(2), 357–372 (1981)
17. Trottenberg, U., Oosterlee, C.W., Schüller, A.: Multigrid. Academic, San Diego (2001)
18. Wallraff, M., Leicht, T., Lange-Hegermann, M.: Numerical flux functions for Reynolds-averaged Navier-Stokes and $k - \omega$ turbulence model computations with a line-preconditioned p-multigrid discontinuous Galerkin solver. Int. J. Numer. Methods Fluids **71**(8), 1055–1072 (2012)
19. Wilcox, D.C.: Reassessment of the scale-determining equation for advanced turbulence models. AIAA J. **26**(11), 1299–1310 (1988)
20. Wilcox, D.C.: Turbulence Modeling for CFD. DCW Industries, Inc., La Canada (1993)

Chapter 6
Taylor Expansion Method for Linear Lattice Boltzmann Schemes with an External Force: Application to Boundary Conditions

François Dubois, Pierre Lallemand, and Mohamed Mahdi Tekitek

Abstract In this contribution we show that it is possible to get the macroscopic fluid equations of a lattice Boltzmann scheme with an external force using the Taylor expansion method. We validate this general expansion by a detailed analysis of the boundary conditions. We derive "quartic parameters" that enforce the precision of the boundary scheme. We explicit and validate the corresponding relations for a Poiseuille flow computed with the D2Q13 lattice Boltzmann scheme (10 December 2013).

6.1 Introduction

The lattice Boltzmann scheme is a mesoscopic method. It deals with a small number of functions that can be interpreted as populations of fictitious particles. It has been developed more than 20 years ago by Qian et al. [19] and Succi et al. [20] among others. This method can simulate various fluid dynamics problems. The dynamics of those particles is such that time, space and momentum are discretized. They move at successive discrete times between the nodes of a regular square lattice. The

F. Dubois (✉)
Laboratoire de Mécanique des Structures et des Systèmes Couplés, CNAM, Paris, France

Department of Mathematics, University Paris Sud, Orsay, France
e-mail: francois.dubois@math.u-psud.fr

P. Lallemand
Beijing Computational Science Research Center, Beijing, China
e-mail: pierre.lal@free.fr

M.M. Tekitek
Faculty of Sciences, University Tunis El Manar, Tunis, Tunisia
e-mail: mohamedmahdi.tekitek@fst.rnu.tn

R. Abgrall et al. (eds.), *High Order Nonlinear Numerical Schemes for Evolutionary PDEs*, Lecture Notes in Computational Science and Engineering 99,
DOI 10.1007/978-3-319-05455-1_6,

velocity space is discretized by a reduced set of discrete velocities. The unknown is the distribution $\{f_i\}$ which is function of velocities, space location and discrete time. In each time step there are two fundamental steps: advection and collision. The advection step is linear and allows the connection of a given vertex with its neighbors. It corresponds to the characteristic method with a Courant number exactly equal to one. The collision step is nonlinear and local in space. The discrete dynamics solve a system of differential equations that converge to an equilibrium state for large time (see the details e.g. in [5]).

A theoretical analysis of the lattice Boltzmann scheme was proposed by Qian et al. [19] and d'Humières [10] with a Chapmann Enskog expansion coming from statistical physics. We refer to Asinari and Ohwada [1] for a method of analysis based on the Grad moment system. A fruitful idea developed by Junk et al. [13] and our team [5] is to use the so-called equivalent equation method derived by Lerat-Peyret [15] and Warming-Hayett [21] in the context of classical finite difference schemes. The time step is considered as an infinitesimal parameter and the finite differences are expanded into a family of equivalent partial differential operators. We observe that the emerging viscosity with the lattice Boltzmann method is in some sense a discrepancy of the scheme that can explicited.

The main goal of this study is to extend the linearized analysis of a lattice Boltzmann scheme using the Taylor expansion method [2, 5] to the case where there is an external body force. In fact few methods [4, 9, 16–18] have been used to incorporate external forces in lattice Boltzmann scheme. In this study, a fraction θ (with $0 \leq \theta \leq 1$) of the force term is added before the collision step and a fraction $(1 - \theta)$ just after. Usually θ is equal to $\frac{1}{2}$. In this contribution we consider a given external force variable in time and space.

Boundary conditions are performed to impose a given velocity (or a given pressure) on a part of the physical boundary. These conditions have to be translated in terms of the lattice Boltzmann scheme. When the scheme is completely defined for internal vertices a numerical boundary conditions of bounce-back or anti-bounce-back type (see e.g. Bouzidi et al. [3]) is equivalent to impose a zero value for some combination of the particle distributions. With the help of the Taylor expansion, it is possible to make in evidence the errors associated to several methods for one dimensional thermal problems or two dimensional fluid problem. This defines so-called "quartic parameters" (see [6]). For the general framework of this kind of methodology, we refer to Ginzburg and Adler [7], Ginzburg et al. [8] and Ginzburg and d'Humières [11]. Our approach is to extend our previous work [6] to three representative examples. In particular a precise choice of relaxation rates allows to spectacularly decrease the error associated to the scheme. In the case of the Poiseuille flow the numerical evaluation of the analytical solution can even be possible with this methodology.

In the first section, we precisely define our numerical scheme. In Sect. 6.3, we extend the linearized analysis of a lattice Boltzmann scheme using the Taylor expansion method [2]. In Sect. 6.4, we study a very simple one dimensional thermal problem with a given temperature on the boundary. The existence of the source term allows an experimental validation of our methodology applied to the

boundary conditions. In Sect. 6.5, we study a two dimensional scheme with the D2Q9 scheme in the context of the Poiseuille flow. This flow is considered with an external force term instead of physical pressure term. Theoretical analysis allows to propose quartic values for the relaxation coefficients and for the parameter θ. These parameters are confronted to a numerical test for this model with 13 velocities in Sect. 6.6. At our knowledge this kind of determination and validation has not been proposed by other teams previously.

6.2 Lattice Boltzmann Scheme with a Forcing Drift

In the following, the notation "DdQq" denotes a lattice Boltzmann scheme with d space dimensions and q velocities. Space is discretized by a regular lattice $\mathscr{L}$ parametrized by a spatial scale Δx. The time step is denoted by Δt. A numerical celerity λ is naturally defined by $\lambda \equiv \frac{\Delta x}{\Delta t}$. The q discrete velocities can be written as $v_j = \lambda\, e_j$ for $0 \leq j \leq q-1$. A classical example is given by the D2Q9 scheme illustrated on the Fig. 6.1.

The lattice Boltzmann scheme is formulated in both spaces of particles and moments m_k, as proposed by d'Humières [10]. We introduce an **invertible** matrix

$$M = (M_{k\,j}), \qquad 0 \leq k,\ j \leq q-1.$$

Then the moments m_k for $0 \leq k \leq q-1$ are defined by linearity:

$$m_k \equiv \sum_{j=0}^{q-1} M_{kj}\ \ f_j.$$

The moment vector $m \in \mathbb{R}^q$ can be written as

$$m = \begin{pmatrix} V \\ Y \end{pmatrix}, \tag{6.1}$$

with two kinds of quantities: conserved moments $V \in \mathbb{R}^N$ and slave moments $Y \in \mathbb{R}^{q-N}$. The conserved moments V are not affected by the collision step when there is no forcing term. If $N = 1$ there is exactly one scalar partial differential equation and when $N > 1$ we obtain an underlying system of N partial differential equations. The family of slave moments Y relax during the collision step towards an equilibrium distribution. An elementary time evolution of a lattice Boltzmann scheme with a force is composed by four steps (see Guo et al. [9]): computation of the dynamical variables, collision (through simple relaxation), forcing of the conserved moments and advection.

The computation of the dynamical variables W from the first component V of the moments (6.1), is given by:

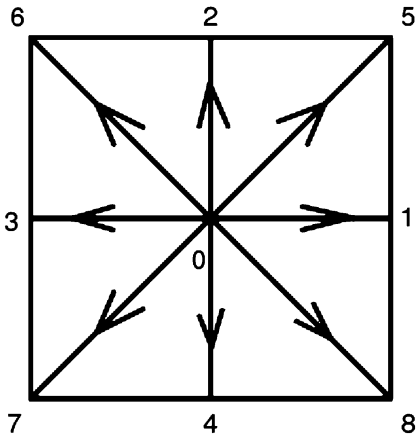


Fig. 6.1 Stencil for the D2Q9 lattice Boltzmann scheme

$$W \;=\; V + \theta\, \Delta t\, F\,, \tag{6.2}$$

where θ is a fixed scalar in $[0, 1]$ and $F \in \mathbb{R}^N$ is a given force term. Remark that the post-processing is obtained from the knowledge of these dynamical variables.

The collision and relaxation step is a redistribution of the populations $\{f_i\}$ at each node x. The N first moments W are not affected by the relaxation.The other "non-conserved" moments relax with a time constant τ_k towards the equilibrium values m_k^{eq}. In this contribution the equilibrium values are a linear function of the dynamical variables W. We suppose that the equilibrium values Y^{eq} of the non-conserved moments are linear function of dynamical variables. We have

$$Y^{eq} \;=\; E\; W\,, \tag{6.3}$$

where E is a fixed matrix with $q - N$ lines and N columns. Thus the relaxation is simply described by an ordinary differential equation:

$$\frac{\mathrm{d}}{\mathrm{d}t}(m_k - m_k^{eq}) + \frac{1}{\tau_k}(m_k - m_k^{eq}) = 0, \quad \text{for} \quad k \geq N.$$

Using an explicit first order Euler scheme, we obtain the algorithm:

$$m_k^* = (1 - s_k)\, m_k + s_k\, m_k^{eq}\,, \tag{6.4}$$

where the superscript $*$ indicates "post-collision" state. The relaxation rates $s_k \equiv \frac{\Delta t}{\tau_k}$ have to satisfy $0 < s_k < 2$ in order to maintain stability of the scheme. With the notation (6.1) the relaxation step is performed in the moment space as follows:

$$Y^* \;=\; (\mathrm{Id} - S)\, Y + S\, Y^{eq}\,, \tag{6.5}$$

where S is the diagonal matrix of the relaxation times s_k for $k \geq N$. We remark that this collision step is local in space.

Due to the force term, the conserved moments during the collision step evolve according to

$$V^* = V + \Delta t\, F = W + (1-\theta)\Delta t\, F. \tag{6.6}$$

If $(\ )^t$ denotes the transpose of the matrix $(\ )$, the moments $m^* \equiv (V^*, Y^*)^{\mathrm{t}}$ after collision are determined by the relations (6.5) and (6.6). After the forcing and collision steps, the particle distribution f_j^* is recovered by the inversion of a linear system of small size:

$$f_j^* = \sum_{k=0}^{q-1} M_{jk}^{-1}\, m_k^*, \quad 0 \leq j \leq q-1. \tag{6.7}$$

The advection step corresponds to a method of characteristics with a Courant number equal to 1 for the advection with velocity v_j. We assume here that for each node x and each velocity v_j, the vertex $x - v_j \Delta t$ is also a node of the lattice. The particles moves from a lattice node x to either itself (for $v_0 = 0$) or one of the $q-1$ neighbors $x_j = x + v_j\, \Delta t$ for velocities $v_j \neq 0$ as presented in the Fig. 6.1. So a time step of a lattice Boltzmann scheme can be written as:

$$f_i(x,\, t+\Delta t) = f_i^*(x - v_i \Delta t,\, t)\,, \quad 0 \leq i \leq q-1\,,\ x \in \mathscr{L}\,, \tag{6.8}$$

where f_i^* denotes the post-collision population of particles evaluated at the end (6.7) of the previous step.

6.3 Taylor Expansion Method for the Equivalent Equations

In this contribution, we extend the "Berliner version" [2] of the Taylor expansion method in order to derive macroscopic equivalent equations when an external forcing term is present. We suppose on one hand that the dynamical variables W satisfy a partial differential equation:

$$\partial_t W = \alpha_1\, W + \Delta t\, \alpha_2\, W + \Delta t^2\, \alpha_3\, W + \cdots + \gamma_0 F + \Delta t\, \gamma_1 F + \ldots \tag{6.9}$$

and on the other hand that the non conserved moments Y follow a dynamical expansion of the type

$$Y = E\, W + \Delta t\, \beta_1\, W + \Delta t^2\, \beta_2\, W + \cdots + \Delta t\, \rho_0 F + \Delta t^2\, \rho_1\, F + \ldots \tag{6.10}$$

In the Eqs. (6.9) and (6.10), α_j, γ_j, β_j and ρ_j are space derivative operators of order j. Moreover α_j and γ_j are $N \times N$ matrices and β_j and ρ_j are $(q - N) \times N$ matrices. Note here that the Eq. (6.9) is a system of N partial differential

equations that represent the evolution of the conserved variables. It gives the macroscopic (equivalent) equation of the physical problem. The Eq. (6.10) describes the expansion of the "slave variables" Y in terms of the conserved quantities. Due to (6.5), we first write the collision step as follows:

$$Y^* = (\mathrm{Id} - S)\, Y + S\, Y^{eq} = (\mathrm{Id} - S)\, Y + S\, E\, W$$

due to (6.3). Then, according to the relation (6.2), we have

$$Y^* = (\mathrm{Id} - S)\, Y + S\, E\, (V + \Delta t\, \theta\, F) = S\, E\, V + (\mathrm{Id} - S)Y + \Delta t\, \theta\, S\, E\, F.$$

Then the moments m^* after forcing and relaxation can be presented as follows:

$$m^* = \begin{pmatrix} V^* \\ Y^* \end{pmatrix} = \begin{pmatrix} I & 0 \\ SE & I - S \end{pmatrix} \begin{pmatrix} V \\ Y \end{pmatrix} + \Delta t \begin{pmatrix} I \\ \theta SE \end{pmatrix} F\,. \tag{6.11}$$

Secondly, we rewrite the scheme (6.8) in moment space:

$$\begin{pmatrix} V \\ Y \end{pmatrix}_k (x, t + \Delta t) = m_k(x, t + \Delta t) = \sum_{j,l} M_{k,j}\; M^{-1}_{j,l}\, m^*_l(x - v_j\, \Delta t, t)$$

then

$$\begin{pmatrix} V \\ Y \end{pmatrix}_k (x, t + \Delta t) = \sum_{j,l,\mu} M_{k,j}\; M^{-1}_{j,l}\, (-v_j)^\mu\, \frac{\Delta t^{|\mu|}}{\mu!}\, \partial_\mu m^*_l(x, t). \tag{6.12}$$

In the previous relation ∂_μ is a space derivation of order $\mu = (\mu_1, \mu_2, \ldots, \mu_d) \in \mathbb{N}^d$ and $\mu! = \mu_1! \ldots \mu_N!$. The length $|\mu|$ of the multi-index μ is equal to $\sum_j \mu_j$. We inject the value (6.11) of m^* in the previous expression. We arrange the previous formal series as increasing powers of Δt. We obtain the following equation:

$$\begin{pmatrix} V \\ Y \end{pmatrix} (t + \Delta t) = \sum_{n \geq 0} \Delta t^n \begin{pmatrix} A_n & B_n \\ C_n & D_n \end{pmatrix} \begin{pmatrix} V \\ Y \end{pmatrix} (t) + \sum_{n \geq 0} \Delta t^{n+1} \begin{pmatrix} G_n \\ H_n \end{pmatrix} F. \tag{6.13}$$

In (6.13), the matrices A_n, B_n, C_n, D_n, G_n and H_n are space derivation operators of order n. They are easy to identify with the help of the right hand side of the Eq. (6.12). The order zero of the development (6.13) relative to Δt is deduced from (6.11) by identification:

$$\begin{pmatrix} A_0 & B_0 \\ C_0 & D_0 \end{pmatrix} = \begin{pmatrix} I & 0 \\ S\, E & \mathrm{I} - S \end{pmatrix}, \quad \begin{pmatrix} G_0 \\ H_0 \end{pmatrix} = \begin{pmatrix} \mathrm{I} \\ \theta\, S\, E \end{pmatrix}. \tag{6.14}$$

Proposition 1 (Taylor expansion at order 1). *The equivalent equations and the non conserved moments for linear lattice Boltzmann scheme with external force up to first order are given by the relations*

$$\partial_t W \;=\; \alpha_1\, W + \gamma_0\, F \;+\; \mathrm{O}(\Delta t)\,, \quad Y \;=\; E\, W + \Delta t\, \big(\beta_1\, W + \rho_0\, F\big) \;+\; \mathrm{O}(\Delta t^2) \tag{6.15}$$

with

$$\begin{cases} \alpha_1 \;=\; A_1 + B_1\, E\,, & \beta_1 \;=\; S^{-1}\,(C_1 + D_1\, E - E\,\alpha_1)\,, \\ \gamma_0 \;=\; G_0 \;=\; \mathrm{I}\,, & \rho_0 \;=\; S^{-1}\,(H_0 - E\,\gamma_0 - \theta\, S\, E)\,. \end{cases} \tag{6.16}$$

Proof. We have from (6.13):

$$\begin{aligned} V(t+\Delta t) &= V + \Delta t\; \partial_t V + \mathrm{O}(\Delta t^2) \\ &= V + \Delta t\; (A_1 V + B_1 Y) + \;\Delta t\; G_0\, F \;+ \mathrm{O}(\Delta t^2) \end{aligned}$$

and due to (6.14),

$$\begin{aligned} Y(t+\Delta t) &= Y + \Delta t\; \partial_t Y + \mathrm{O}(\Delta t^2) \\ &= (\mathrm{Id} - S)\, Y + S\, E\, V + \Delta t\; (C_1 V + D_1\, Y) + \Delta t\; H_0\, F + \mathrm{O}(\Delta t^2). \end{aligned}$$

Using the relation (6.2) we have

$$V \;=\; W - \theta \Delta t F$$

and the development at equilibrium (order zero)

$$Y \;=\; E\, W + \mathrm{O}(\Delta t),$$

we obtain on one hand

$$\partial_t W \;=\; (A_1 + B_1 E)\, W + G_0 F \;+\; \mathrm{O}(\Delta t)$$

and on the other hand

$$Y \;=\; EW + \Delta t\; S^{-1}\,[C_1 + D_1 E - E\,(A_1 + B_1 E)\, W] + (H_0 - E\gamma_0 - \theta\; S\; E)\; F + \mathrm{O}(\Delta t^2).$$

By identification of the previous relations with the ansatz (6.9) and (6.10), we obtain the relations (6.15) at first order with the coefficients α_1, β_1, γ_0 and ρ_0 explicited in the relations (6.16). □

Proposition 2 (Taylor expansion at order 2). *The equivalent equations for linear lattice Boltzmann scheme with external force, as described in first section, up to second order, are:*

$$\left\{\begin{array}{l} \partial_t W \,=\, \alpha_1\, W + \gamma_0\, F + \Delta t\, \big(\alpha_2\, W \,+\, \gamma_1\, F\big) \,+\, \mathrm{O}(\Delta t^2) \\ \\ \quad Y \,=\, E\, W + \Delta t\, \big(\beta_1\, W + \rho_0\, F\big) + \Delta t^2\, \big(\beta_2\, W + \rho_1\, F\big) \,+\, \mathrm{O}(\Delta t^3) \end{array}\right. \tag{6.17}$$

with coefficients α_1, β_1, γ_0 *and* ρ_0 *proposed in (6.16) and*

$$\left\{\begin{array}{l} \alpha_2 \,=\, B_1\, \beta_1 + A_2 + B_2\, E - \frac{1}{2}\alpha_1^2 \\ \\ \gamma_1 \,=\, B_1\, \rho_0 + G_1 - \theta\, A_1 - \frac{1}{2}\big[\alpha_1\, \gamma_0 + (\gamma_0 - 2\theta)\, \partial_t\big] \\ \\ \beta_2 \,=\, S^{-1}\big[D_1\, \beta_1 - E\, \alpha_2 - \beta_1\, \alpha_1 - E\, \frac{\alpha_1^2}{2} + C_2 + D_2\, E\big] \\ \\ \rho_1 \,=\, S^{-1}\big[D_1 \rho_0 + H_1 - E\, \gamma_1 - \beta_1\, \gamma_0 - \frac{1}{2} E \alpha_1\, \gamma_0 \\ \qquad\qquad -\frac{1}{2} E\, \gamma_0\, \partial_t - \rho_0\, \partial_t - \theta\, C_1\big]. \end{array}\right. \tag{6.18}$$

Proof. We first differentiate relative to time the first relation of (6.15):

$$\begin{aligned} \partial_t^2 W \,&=\, \partial_t\big(\alpha_1\, W + F\big) + \mathrm{O}(\Delta t) = \alpha_1\, \big(\alpha_1\, W + F\big) + \,\partial_t F + \mathrm{O}(\Delta t) \\ &= \alpha_1^2\, W + (\alpha_1 + \partial_t)\, F + \mathrm{O}(\Delta t). \end{aligned}$$

We can therefore develop the dynamical variables W at time $t + \Delta t$ up to second order. Due to the definition (6.2), we have:

$$\begin{aligned} W(t + \Delta t) \,&=\, V(t + \Delta t) + \theta\, \Delta t\, F(t + \Delta t) \\ \\ &= V(t) + \Delta t\, \big(A_1\, V(t) + B_1\, Y(t)\big) + \Delta t^2\, \big(A_2\, V(t) + B_2\, Y(t)\big) \\ &\quad + \theta\, \Delta t\, F(t + \Delta t) + \,\mathrm{O}(\Delta t^3) \\ \\ &= W(t) - \theta\, \Delta t\, F(t) + \theta\, \Delta t\, F(t + \Delta t) \,+\, \Delta t\, \big(A_1\, (W - \theta\, \Delta t\, F) \\ &\quad + B_1\, (E\, W + \Delta t\, \beta_1\, W + \Delta t\, \rho_0\, F)\big) \\ &\quad + \Delta t^2\, \big(A_2\, W + B_2\, E\, W\big) + \mathrm{O}(\Delta t^3)\,. \end{aligned}$$

Then taking into account the Taylor formula at second order accuracy and the expression of $\partial_t^2 W$:

$$\begin{aligned} &W + \Delta t\, \partial_t W + \frac{\Delta t^2}{2}\big[\alpha_1^2\, W + (\alpha_1 + \partial_t)\, F\big] + \mathrm{O}(\Delta t^3) \\ &\quad =\, W + \theta\, \Delta t^2\, \partial_t\, F \,+\, \Delta t\, \big(A_1\, W + B_1\, E\, W\big) \\ &\qquad + \Delta t^2\, \big(-\,\Delta t\, F + B_1\, \beta_1\, W + B_1\, \rho_0\, F + A_2\, W + B_2\, E\, W\big) + \mathrm{O}(\Delta t^3). \end{aligned}$$

After dividing by Δt:

$$\begin{aligned}\partial_t W = (A_1 + B_1\, E)\, W + F + \Delta t \left[A_2 + B_2\, E + B_1\, \beta_1 - \tfrac{1}{2}\alpha_1^2\right] W \\ + \Delta t \left[G_1 + B_1\, \rho_0 - \theta\, A_1 - \tfrac{1}{2}\alpha_1 + (\theta - \tfrac{1}{2})\, \partial_t\right] F + \mathrm{O}(\Delta t^2)\end{aligned}$$

and the expressions of α_2 and γ_1 in (6.18) are established.

To get the second relation of (6.17), we first look to the derivation relative to time of the non-conserved moments Y. We have

$$\begin{aligned}\partial_t Y &= \partial_t \left[E\, W + \Delta t \left(\beta_1\, W + \rho_0\, F\right)\right] + \mathrm{O}(\Delta t^2) \\ &= E \left[\alpha_1\, W + F + \Delta t\, (\alpha_2\, W + \gamma_1\, F) + \mathrm{O}(\Delta t^2)\right] \\ &\quad + \Delta t\, \beta_1 \left(\alpha_1\, W + F\right) + \Delta t\, \rho_0\, \partial_t\, F + \mathrm{O}(\Delta t^2) \\ &= E(\alpha_1\, W + F) + \Delta t[(E\, \alpha_2 + \beta_1\, \alpha_1) W + (E\, \gamma_1 + \beta_1 + \rho_1\, \partial_t) F] + \mathrm{O}(\Delta t^2).\end{aligned}$$

We differentiate relative to time at second order:

$$\begin{aligned}\partial_t^2 Y &= \partial_t \left[E \left(\alpha_1\, W + F\right)\right] + \mathrm{O}(\Delta t) = E\, \alpha_1 \left(\alpha_1\, W + F\right) + E\, \partial_t\, F + \mathrm{O}(\Delta t) \\ &= E\, \alpha_1^2\, W + (E\, \alpha_1 + E\, \partial_t)\, F + \mathrm{O}(\Delta t).\end{aligned}$$

Then we have

$$\begin{aligned}S\, Y(t) &= S\, E\, V - \big(Y(t + \Delta t) - Y(t)\big) + \Delta t\, (C_1\, V + D_1\, Y + \theta\, S\, E\, F) \\ &\quad + \Delta t^2\, (C_2\, V + D_2\, Y + H_1\, F) + \mathrm{O}(\Delta t^3) \\ &= S\, E\, (W - \theta\, \Delta t\, F) \\ &\quad - \Delta t \big[E \left(\alpha_1\, W + F\right) + \Delta t \big[(E\, \alpha_2 + \beta_1\, \alpha_1) W + (E\, \gamma_1 + \beta_1 + \rho_1\, \partial_t) F\big]\big] \\ &\quad - \tfrac{\Delta t^2}{2} \left[E\, \alpha_1^2\, W + (E\, \alpha_1 + E\, \partial_t)\, F\right] \\ &\quad + \Delta t \big[C_1 (W - \theta\, \Delta t\, F) + D_1 (E\, W + \Delta t\, \beta_1\, W + \Delta t\, \rho_0\, F) + \theta\, S\, E\, F\big] \\ &\quad + \Delta t^2\, (C_2\, W + D_2\, E\, W + H_1\, F) + \mathrm{O}(\Delta t^3) \\ &= S \left[E\, W + \Delta t\, \beta_1\, W + \Delta t\, \rho_0\, F\right] \\ &\quad + \Delta t^2 \left[\big(C_2 + D_2\, E + D_1\, \beta_1 - \tfrac{1}{2} E\, \alpha_1^2 - (E\, \alpha_2 + \beta_1\, \alpha_1)\big) W\right. \\ &\quad \left. + \big(H_1 + D_1\, \rho_0 - \theta\, C_1 - \tfrac{1}{2}(E\, \alpha_1 + E\, \partial_t) - (E\, \gamma_1 + \beta_1 + \rho_0\, \partial_t)\big)\, F\right] \\ &\quad + \mathrm{O}(\Delta t^3)\end{aligned}$$

and the expressions of β_2 and ρ_1 proposed at the second line of the relations (6.18) are established because γ_0 is the identity matrix. □

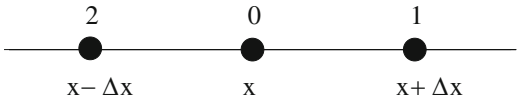


Fig. 6.2 Stencil of the lattice Boltzmann scheme D1Q3

6.4 Boundary Conditions for D1Q3 Thermal Problems

In general there is a difference between the assumed wall position and the "measured" one, where the numerical solution vanishes. For some particular lattice Boltzmann parameters, this difference is null up to order two on Δx. In this section, we study the impact of the knowledge given in the relations (6.17) and (6.18) on the analysis of the boundary conditions for the lattice Boltzmann schemes. In general, the incoming particles are given as a simple function of the outgoing ones through specular reflection (bounce-back) [3] or specular "anti-reflection" (anti-bounce-back) that are detailed in the corpus of the text. We first focus on the D1Q3 scheme for a scalar problem.

We introduce the following moment matrix M for the D1Q3 (see Fig. 6.2) lattice Boltzmann scheme:

$$M = \begin{pmatrix} 1 & 1 & 1 \\ 0 & \lambda & -\lambda \\ -2\,\lambda^2 & \lambda^2 & \lambda^2 \end{pmatrix}. \tag{6.19}$$

We consider only one conservation law ($N = 1$) and we set $\rho = W = f_0 + f_1 + f_2$. The equilibrium values Y^{eq} of the two non conserved moments are given according to

$$Y^{eq} = \begin{pmatrix} 0 \\ \lambda^2 \\ \zeta \end{pmatrix} \rho \equiv E\ \rho. \tag{6.20}$$

Proposition 3 (Diffusion model with the D1Q3 lattice Boltzmann scheme). *With the previous lattice Boltzmann D1Q3 scheme, described by relations (6.2), (6.6), (6.8), (6.19) and (6.20), we have the equivalent equation*

$$\frac{\partial \rho}{\partial t} - \kappa\, \frac{\partial^2 \rho}{\partial x^2} - F = \mathrm{O}(\Delta t^2)$$

at second order for a stationary force F, with the thermal diffusivity

$$\kappa = \frac{2+\zeta}{3}\, \Delta t\, \lambda^2\, \sigma_1 \qquad \textit{with} \qquad \sigma_1 \equiv \frac{1}{s_1} - \frac{1}{2}.$$

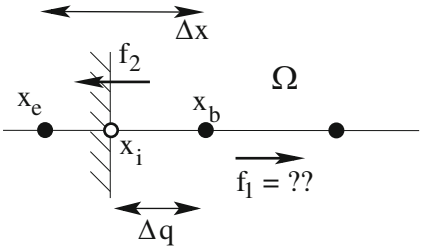


Fig. 6.3 "Anti-bounce-back" numerical boundary condition for the D1Q3 lattice Boltzmann scheme. The boundary node x_b is located at the distance Δq of the theoretical boundary. The incoming particles $f_1(x_b)$ are determined from the knowledge of the outgoing density of particles $f_2(x_b)$

Proof. There is only one scalar conservation and $W \equiv \rho$. According to the Proposition 2, we have the equivalent equation up to order two:

$$\partial_t W = \alpha_1 W + F + \Delta t \left(\alpha_2 W + \gamma_1 F\right) + \mathrm{O}(\Delta t^2).$$

The associated coefficients can be computed easily: $\alpha_1 = 0, \alpha_2 = \lambda^2 \sigma_1 \frac{\zeta+2}{3}$, $\gamma_0 = 1$ and $\gamma_1 = (\theta - \frac{1}{2})\,\partial_t$. We have in consequence $\gamma_1 F = 0$ if $\partial_t F = 0$ and the equivalent equation is established. □

We consider the elementary one-dimensional Poisson problem with homogeneous Dirichlet boundary conditions:

$$-\Delta u(x) = c \quad \text{for } 0 < x < 1\,, \qquad u(0) = u(1) = 0. \tag{6.21}$$

The solution of the problem (6.21) is quadratic. A uniform body force F is considered to take into account the source term c in the right hand side of the Poisson equation. We denote by x_b the first node at the left in the domain $]0,\ 1[$ and by x_e a fictitious outside node (see Fig. 6.3). The node x_b is supposed to be located at a distance Δq of the solid wall position x_i (see Fig. 6.3). We implement a so-called "anti-bounce-back" boundary condition to take into account the Dirichlet boundary conditions:

$$f_1(x_b, t + \Delta t) = -f_2^*(x_b, t) + \Phi \tag{6.22}$$

where Φ is a given boundary term directly related to the boundary data. Observe that this term is null if we consider e.g. homogenous Dirichlet boundary conditions.

Proposition 4 (Quartic treatment of a D1Q3 boundary condition). *For the above D1Q3 lattice Boltzmann scheme, we define*

$$\xi \equiv -3 + 8\sigma_1\sigma_2 - 8\sigma_1\,(1 - 2\theta).$$

The numerical solution of the Poisson problem (6.21) vanishes at third order formal accuracy for the position $\Delta q = \frac{\Delta x}{2}$ if the condition $\xi = 0$ is satisfied.

Proof. Using the Eq. (6.10) on the non conserved moment, we have

$$Y = E\,W + \Delta t\;\beta_1 W + \Delta t^2\;\beta_2 W + \Delta t\;\rho_0\,F + \Delta t^2\;\rho_1\,F.$$

Then we perform a collision step to get

$$Y^* = (\mathrm{I} - S)\,Y + S\,E\,Y + \Delta t\;\theta\;S\;E\;F.$$

With the help of the inverse moments matrix M^{-1}, we deduce:

$$f_1^* = \frac{1}{2\lambda^2}\,[2m_2^* + \lambda\,m_1^*] \qquad \text{and} \qquad f_2^* = \frac{1}{2\lambda^2}\,[2m_2^* - \lambda\,m_1^*].$$

We impose the anti-bounce-back condition on the boundary:

$$f_1(x_b,\,t + \Delta t) = -f_2(x_e,\,t + \Delta t) + \Phi.$$

Due to the fundamental time iteration of a lattice Boltzmann scheme (6.8), we can write this relation under the form

$$f_1^*(x_e, t) = -f_2^*(x_b, t) + \Phi.$$

Using a Taylor development of the moments, we obtain:

$$\begin{aligned} f_1^*(x_e) + f_2^*(x_b) = {} & \tfrac{(2+\zeta)}{3}\rho(x_i) + \\ & + \Delta t^2\lambda^2\tfrac{(2+\zeta)}{72}\big[-3 + 8\sigma_1\sigma_2 - 8\sigma_1(1-2\theta)\big]\tfrac{\partial^2\rho}{\partial x^2}(x_i) + \mathrm{O}(\Delta t^3). \end{aligned}$$

We can precise that

$$\Phi = \frac{(2+\zeta)}{3}\rho(x_i)$$

in the right hand side of (6.22). Then the conclusion holds if we give a null value for the second order term in the previous relation. □

Remark. We have extended our result [6] for a general values of the parameter θ (Figs. 6.4 and 6.5).

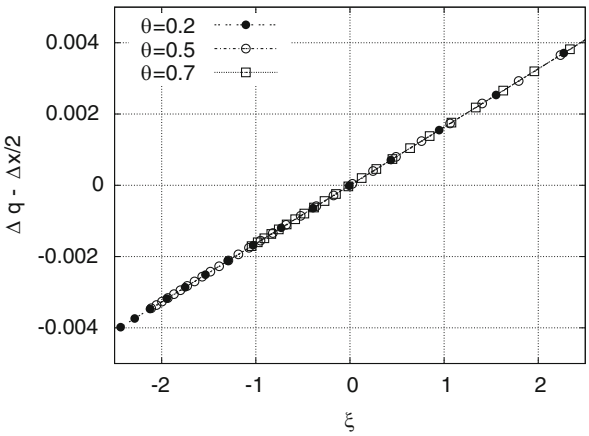


Fig. 6.4 Results of simulations for a the solid wall position Δq vs. the parameter $\xi \equiv -3 + 8\sigma_1\sigma_2 - 8\sigma_1(1-2\theta)$ for the D1Q3 scheme for thermal model. We have considered variations of the parameter θ of the relation (6.2) for the forcing term ($\theta = 0.2$ with *bullets*, $\theta = 0.5$ with *circles* and $\theta = 0.7$ with *boxes*) and of the parameter σ_2 relative to the last moment

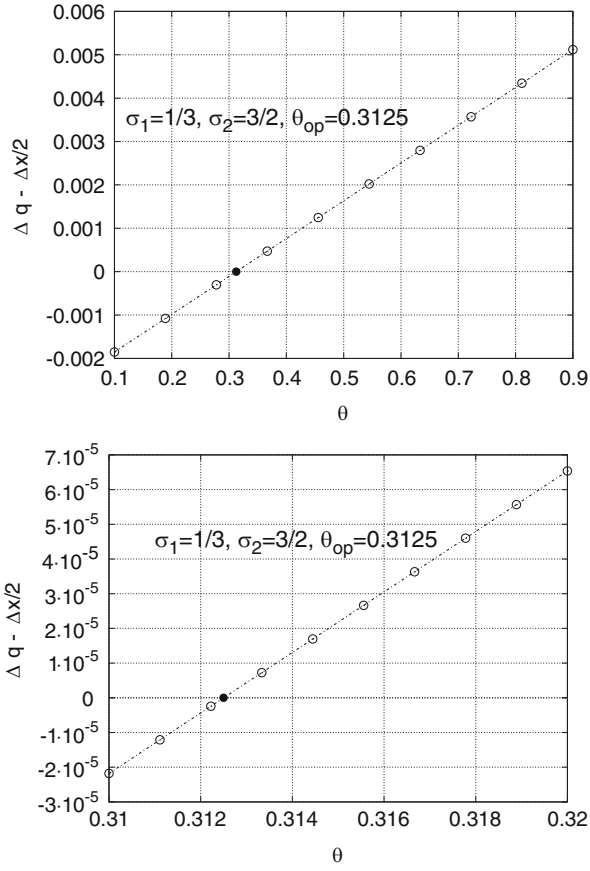


Fig. 6.5 Analogous to the Fig. 6.4. On the *top*: regular variation of the parameter θ for $\theta \in [0.1, 0.9]$. *Bottom*: zoom of the top figure for $\theta \in [0.31, 0.32]$

6.5 D2Q9 for a Linear Acoustic Type Fluid

This scheme is described and analyzed in [14]. The matrix M is now 9×9. For a fluid model, we have $N = 3$ conserved quantities: density $m_0 \equiv \rho$ and the two components $m_1 \equiv J_x$ and $m_2 \equiv J_y$ of the momentum. The non conserved moments m_k for $k \geq 3$ are detailed in [14]. Their equilibrium values are classical:

$$m_3^{eq} = -2\rho, \quad m_4^{eq} = \rho, \quad m_5^{eq} = -\frac{J_x}{\lambda}, \quad m_6^{eq} = -\frac{J_y}{\lambda}, \quad m_7^{eq} = m_8^{eq} = 0.$$

The relaxation rates are labelled from s_3 to s_8 with the two constraints $s_5 = s_6$ and $s_7 = s_8$ to recover an isotropic model compatible with the physics. The particle directions are represented on the Fig. 6.1. With a forcing term of the type $F = (0,\ F_x,\ 0)$, the partial equivalent equations at the order 2 are given by

$$\left\{ \begin{array}{l} \dfrac{\partial \rho}{\partial t} + \dfrac{\partial J_x}{\partial x} + \dfrac{\partial J_y}{\partial y} = \mathrm{O}(\Delta t^2)\,, \\ \dfrac{\partial J_x}{\partial t} + c_s^2 \dfrac{\partial \rho}{\partial x} - \zeta \dfrac{\partial}{\partial x}\left[\dfrac{\partial J_x}{\partial x} + \dfrac{\partial J_y}{\partial y}\right] - \nu\, \Delta J_x - F_x = \mathrm{O}(\Delta t^2)\,, \\ \dfrac{\partial J_y}{\partial t} + c_s^2 \dfrac{\partial \rho}{\partial y} - \zeta \dfrac{\partial}{\partial y}\left[\dfrac{\partial J_x}{\partial x} + \dfrac{\partial J_y}{\partial y}\right] - \nu\, \Delta J_y = \mathrm{O}(\Delta t^2), \end{array} \right. \tag{6.23}$$

where sound speed c_s, the shear viscosity ν and the bulk viscosity ζ satisfy respectively $c_s^2 = \frac{\lambda}{3}$, $\nu = \Delta t\, \sigma_7\, \frac{\lambda^2}{3}$ and $\zeta = \Delta t\, \sigma_3\, \frac{\lambda^2}{3}$ for the D2Q9 scheme.

We consider the two-dimensional Poiseuille flow with the D2Q9 lattice Boltzmann scheme. We model this classical problem with the help of an imposed volumic force. Let $\Omega = [1, N_x] \times [1, N_y]$. We enforce periodic condition at the inlet ($i = 1$) and at the outlet ($i = N_x$). A "bounce-back" boundary condition is imposed on the walls to take into account the null velocity $J_x = 0$ on the upper and lower boundary. A uniform body force F_x is applied to model the pressure gradient. We observe that for stationary problems, a lot of time iterations are necessary in order to obtain correct converged results. In that case the lattice Boltzmann method is not the most efficient taking in to account the fact that due the exact advection of the lattice the resolution of the stationary problem is not a simple task.

Proposition 5 (Precise position of the Poiseuille boundary with D2Q9). *With the above choices for the lattice Boltzmann scheme, the solid wall is located at* $\Delta q = \frac{\Delta x}{2}$ *up to third order when the parameter*

$$\xi \equiv -3 + 8\sigma_8\sigma_5 - 8\sigma_8(1 - 2\theta)$$

is equal to zero.

The proof of the above Proposition is elementary with traditional algebraic methods (see e.g. [6]) but quite long to develop. We have implemented it without difficulty with a formal computer software. In order to convince the reader, we test the resulting condition $\xi = 0$ with appropriate numerical experiments presented in Figs. 6.6 and 6.7. These experiments extend the ones presented in [6].

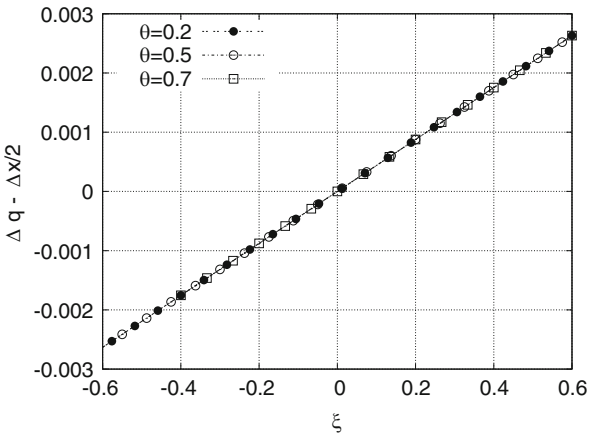


Fig. 6.6 Poiseuille flow for the D2Q9 scheme. Solid wall position Δq vs. the parameter $\xi \equiv -3 + 8\sigma_8\sigma_5 - 8\sigma_8(1-2\theta)$. Variations of the parameter θ of the relation (6.2) for the forcing term ($\theta = 0.2$ with *bullets*, $\theta = 0.5$ with *circles* and $\theta = 0.7$ with *boxes*) and of the parameter σ_5

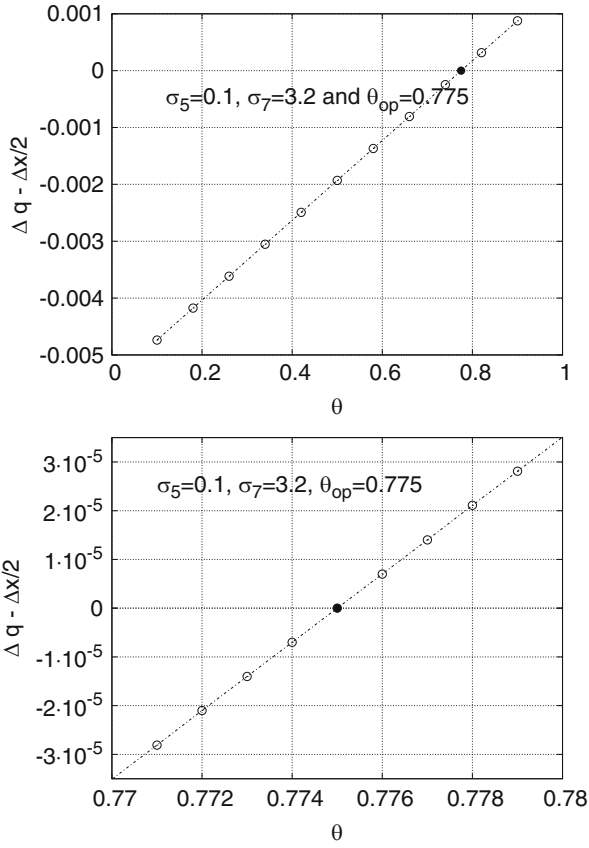


Fig. 6.7 Poiseuille flow with the D2Q9 lattice Boltzmann scheme. *Top*: solid wall position Δq vs. the parameter $\xi \equiv -3 + 8\sigma_8\sigma_5 - 8\sigma_8(1-2\theta)$. Variations of the parameter θ of the relation (6.2) for the forcing term ($\theta = 0.2$ with *bullets*, $\theta = 0.5$ with *circles* and $\theta = 0.7$ with *boxes*) and of the parameter σ_5 relative of the moment m_5. *Bottom*: zoom of the top figure, with a variation Δq vs. the parameter $\theta \in [0.77,\ 0.78]$

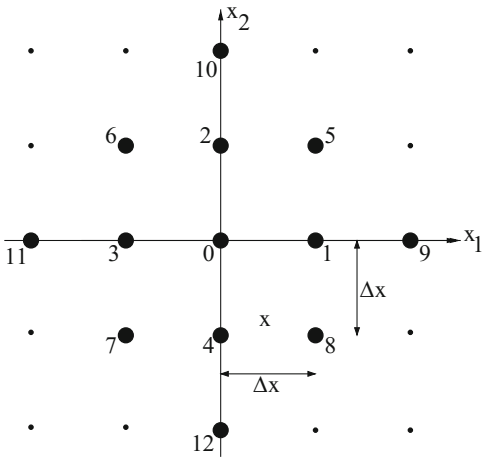


Fig. 6.8 Stencil of the D2Q13 lattice Boltzmann scheme

6.6 D2Q13 for a Linearized Fluid

To construct the scheme D2Q13, we add four more velocities to the previous scheme D2Q9 (see the Fig. 6.8). The details can be found e.g. in Higuera et al. [12]. Nine moments are analogous to those proposed previously. The equilibrium values are parameterized according to

$$\left\{ \begin{array}{c} m_3^{eq} = a_1\,\rho,\ m_4^{eq} = m_5^{eq} = 0,\ m_6^{eq} = \frac{c_1}{\lambda} J_x,\ m_7^{eq} = \frac{c_1}{\lambda} J_y, \\ m_8^{eq} = -\frac{65+63\,c_1}{24\,\lambda} J_x,\ m_9^{eq} = -\frac{65+63\,c_1}{24\,\lambda} J_y,\ m_{10}^{eq} = a_2\,\rho,\ m_{11}^{eq} = a_3\,\rho,\ m_{12}^{eq} = 0. \end{array} \right.$$

In order to simulate isotropic models, the relaxation rates s_3 to s_{12} satisfy the following relations: $s_4 = s_5$, $s_6 = s_7$ and $s_8 = s_9$ as proposed e.g. in [2]. With the previous choices, with a forcing term $F = (0,\ F_x,\ 0)$ analogous to the one proposed previously for the D2Q9 scheme, the partial equivalent equations take the form (6.23). In this case the sound speed c_s, the shear viscosity ν and the bulk viscosity ζ satisfy respectively $c_s^2 = \frac{28+a_1}{26}\,\lambda$, $\nu = \Delta t\,\sigma_5\,\frac{\lambda^2}{2}$ and $\zeta = -\Delta t\,\sigma_3\,\frac{2+a_1}{26}\,\lambda^2$. In our computation we set $a_1 = -12$. Then $\zeta > 0$. We study a Poiseuille flow with the D2Q13 lattice Boltzmann scheme in the framework of an imposed source term. The general choices are analogous to the ones we did with the D2Q9 lattice Boltzmann scheme.

Proposition 6 (Precise position of the Poiseuille boundary with D2Q13). *When the parameter*

$$\xi \equiv -7 + 40\sigma_5\,\sigma_7 - 8\sigma_5\,(1 - 2\theta)$$

is null and with the above choices for the lattice Boltzmann scheme, the solid wall is located at $\Delta q = \frac{\Delta x}{2}$ *at third order accuracy if and only if* $\xi = 0$.

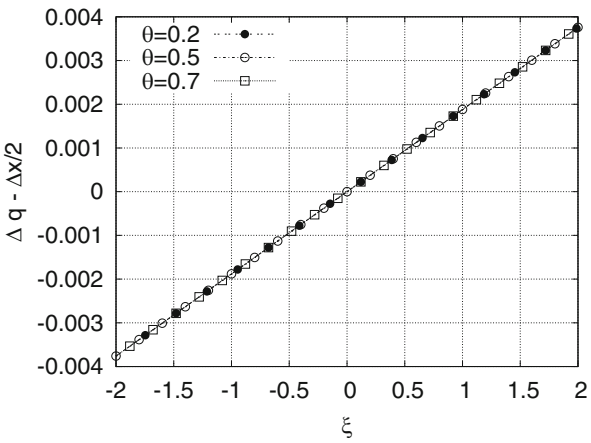


Fig. 6.9 Poiseuille flow for the D2Q13 scheme. Solid wall position Δq vs. the parameter $\xi \equiv -7 + 40\sigma_5\,\sigma_7 - 8\sigma_5\,(1-2\theta)$. Variations of the parameter θ of the relation (6.2) for the forcing term ($\theta = 0.2$ with *bullets*, $\theta = 0.5$ with *circles* and $\theta = 0.7$ with *boxes*) and of the parameter σ_5

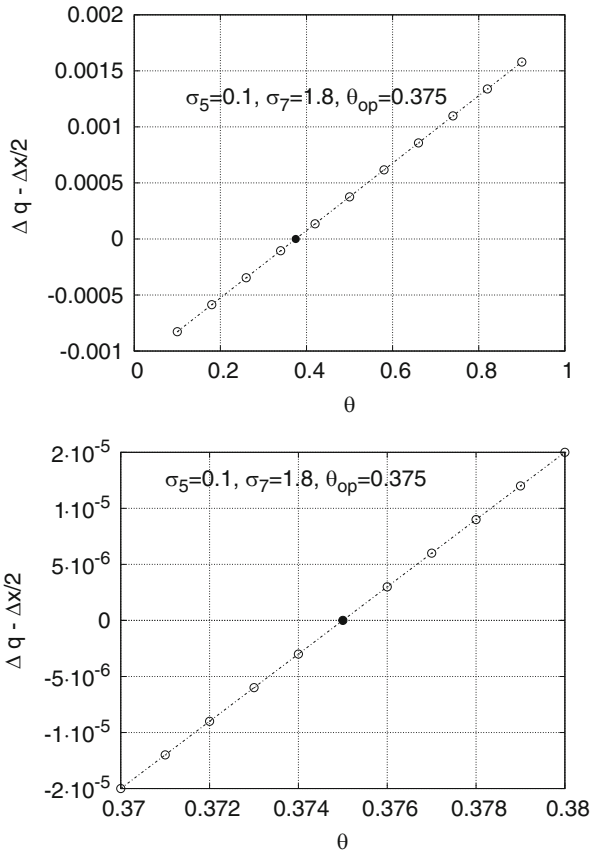


Fig. 6.10 Precise Poiseuille flow with the D2Q13 lattice Boltzmann scheme. Solid wall position Δq vs. the parameter θ for the forcing term in (6.2). *Top*: $\theta \in [0.1, 0.9]$, *bottom*: zoom of the top figure with $\theta \in [0.37, 0.38]$

This proposition has been solved with the help of formal calculus. We did various experiments analogous to those presented for the D2Q9 scheme. They are summarized in the captions of Figs. 6.9 and 6.10.

6.7 Conclusion

In this contribution, we have extended the "Berliner version" [2] of the formal Taylor expansion method for linear lattice Boltzmann scheme to the case of a possible external force. We have established that the quartic parameters to enforce the precision of the lattice Boltzmann scheme depend not only on the relaxation rates but also on the choice of the parameter θ for the time integration (6.2) of the source term. We have derived quartic parameters for the D2Q13 model for a Poiseuille flow. Our numerical results validate the formal Taylor development. Nevertheless, a rigorous numerical analysis is still an open question. In future work, we wish to extend the quest of quartic parameters for more general physical problems, study time dependent boundary conditions and give general boundary conditions for any position of the wall relative to the lattice.

Acknowledgements The authors thanks the referee for helpful questions after the first draft of this contribution.

References

1. Asinari, P., Ohwada, T.: Connection between kinetic methods for fluid-dynamic equations and macroscopic finite-difference schemes. Comput. Math. Appl. **58**, 841–861 (2009)
2. Augier, A., Dubois, F., Graille, B., Lallemand, P.: On rotational invariance of lattice Boltzmann schemes. Comput. Math. Appl. **67**(2), 239–255 (2014)
3. Bouzidi, M., Firdaous, M., Lallemand, P.: Momentum transfer of a Boltzmann-lattice fluid with boundaries. Phys. Fluids **13**, 3452–3459 (2001)
4. Buick, J.M., Greated, C.A.: Gravity in a lattice Boltzmann model. Phys. Rev. **61**, 5307–5320 (2000)
5. Dubois, F.: Equivalent partial differential equations of a lattice Boltzmann scheme. Comput. Math. Appl. **55**, 1441–1449 (2008)
6. Dubois, F., Lallemand, P., Tekitek, M.M.: On a superconvergent lattice Boltzmann boundary scheme. Comput. Math. Appl. **59**, 2141–2149 (2010)
7. Ginzburg, I., Adler, P.M.: Boundary flow condition analysis for the three-dimensional lattice Boltzmann model. J. Phys. II **4**, 191–214 (1994)
8. Ginzburg, I., Verhaeghe, F., d'Humières, D.: Two-relaxation-time lattice Boltzmann scheme: about parametrization, velocity, pressure and mixed boundary conditions. Commun. Comput. Phys. **3**, 519–581 (2008)
9. Guo, Z., Zheng, C., Shi, B.: Discrete lattice effects on the forcing term in the lattice Boltzmann method. Phys. Rev. E **65**, 1–6 (2002)
10. d'Humières, D.: Generalized Lattice-Boltzmann equations. In: Shizgal, B.D., Weave, D.P. (eds.) Rarefied Gas Dynamics: Theory and Simulations. AIAA Progress in Astronautics and Aeronautics, vol. 159, pp. 450–458. AIAA, Washington, D.C. (1992)

11. d'Humières, D., Ginzburg, I.: Multi-reflection boundary conditions for lattice Boltzmann models. Phys. Rev. E **68**, 066614 (2003)
12. Higuera, F.J., Succi, S., Benzi, R.: Lattice gas dynamics with enhanced collisions. EuroPhys. Lett. **9**, 345–349 (1989)
13. Junk, M., Klar, A., Luo, L.-S.: Asymptotic analysis of the lattice Boltzmann equation. J. Comput. Phys. **210**, 676–704 (2005)
14. Lallemand, P., Luo, L.-S.: Theory of the lattice Boltzmann method: dispersion, dissipation, isotropy, Galilean invariance, and stability. Phys. Rev. E **61**, 6546–6562 (2000)
15. Lerat, A., Peyret, R.: Noncentered schemes and shock propagation problems. Comput. Fluids **2**, 35–52 (1974)
16. Luo, L.-S.: Unified theory of lattice Boltzmann models for non-ideal gases. Phys. Rev. Lett. **81**, 1618–1621 (1998)
17. Martys, N.S., Shan, X., Chen, H.: Evaluation of the external force term in the discrete Boltzmann equation. Phys. Rev. E **58**, 6855–6857 (1998)
18. Mohamad, A.A., Kuzmin, A.: A critical evaluation of force term in lattice Boltzmann method, natural convection problem. Int. J. Heat Mass Transf. **53**, 990–996 (2010)
19. Qian, Y.H., d'Humières, D., Lallemand, P.: Lattice BGK models for Navier-Stokes equation. EuroPhys. Lett. **17**, 479–484 (1992)
20. Succi, S., Benzi, R., Higuera, F.: Lattice gas methods for homogeneus and inhomegeneus hydrodynamics. In: Monaco, R. (ed.) Discrete Kinetic Theory, Lattice Gas Dynamics and Foundations of Hydrodynamics, pp. 329–342. World Scientific, Singapore (1989)
21. Warming, R.F., Hyett, B.J.: The modified equation approach to the stability and accuracy analysis of finite difference methods. J. Comput. Phys. **14**, 159–179 (1974)

Chapter 7
High Order SFV and Mixed SDG/FV Methods for the Uncertainty Quantification in Multidimensional Conservation Laws

Svetlana Tokareva, Christoph Schwab, and Siddhartha Mishra

Abstract We analyze and apply the high order Stochastic Finite Volume (SFV) and mixed Stochastic Discontinuous Galerkin/Finite Volume (SDG/FV) methods used to quantify the uncertainty in hyperbolic conservation laws with random initial data and flux coefficients. We describe incomplete information in the conservation law mathematically as random fields. The SFVM is formulated to solve numerically the system of conservation laws with sources of randomness in both flux coefficients and initial data. We formulate the Stochastic Discontinuous Galerkin method which we primarily use to solve the multidimensional stochastic conservation laws on unstructured grids. Finally, we compare the efficiency of the SFV and SDG methods with of Monte-Carlo type methods. Finally, we introduce an adaptation technique based on the Karhunen-Loève expansion of the random flux and/or initial data and apply it in order to reduce the computational cost of the SFVM.

7.1 Introduction

We analyze and apply the high order Stochastic Finite Volume method (SFVM) [1–3, 20] and mixed Stochastic Discontinuous Galerkin/Finite Volume (SDG/FV) methods used to quantify the uncertainty in hyperbolic conservation laws with random initial data and flux coefficients. The direct application of these methods

S. Tokareva (✉)
ASCOMP GmbH, Zurich, Switzerland

Seminar for Applied Mathematics, ETH Zurich, Zurich, Switzerland
e-mail: tokareva@ascomp.ch

C. Schwab • S. Mishra
Seminar for Applied Mathematics, ETH Zurich, Zurich, Switzerland

R. Abgrall et al. (eds.), *High Order Nonlinear Numerical Schemes for Evolutionary PDEs*, Lecture Notes in Computational Science and Engineering 99,
DOI 10.1007/978-3-319-05455-1_7,

is the simulation of gas flows with uncertain physical parameters and/or initial flow conditions. Many efficient numerical methods have been developed to approximate the entropy solutions of systems of conservation laws [7], however, in many practical applications it is not always possible to obtain exact physical data due to, for example, measurement or modeling errors. We describe incomplete information in the conservation law mathematically as random fields. Such data are described in terms of statistical quantities of interest like the mean, variance, higher statistical moments; in some cases the distribution law of the stochastic initial data is also assumed to be known. There exist several techniques to quantify the uncertainty (i.e. determine the mean flow and its statistical moments), such as the Monte-Carlo (MC), the Multi-Level Monte Carlo (MLMC) and Stochastic Galerkin method (see also [5, 8–15, 19, 22–26]). The SFVM is formulated to solve numerically the system of conservation laws with sources of randomness in both flux coefficients and initial data. We generalize this approach and formulate the Stochastic Discontinuous Galerkin method which we primarily use to solve the multidimensional stochastic conservation laws on unstructured grids. Finally, we compare the efficiency of the SFV and SDG methods with of Monte-Carlo type methods. Finally, we introduce an adaptation technique based on the Karhunen-Loève expansion of the random flux and/or initial data and apply it in order to reduce the computational cost of the SFVM.

7.2 Conservation Laws with Random Flux and Initial Data

Consider the hyperbolic system of conservation laws with random flux coefficients

$$\partial_t \mathbf{U} + \nabla_x \cdot \mathbf{F}(\mathbf{U}, \omega) = \mathbf{0}, \ t > 0; \tag{7.1}$$

$\mathbf{x} = (x_1, x_2, x_3) \in D_x \subset \mathbb{R}^3$, $\mathbf{U} = [u_1, \ldots, u_p]^T$, $\mathbf{F} = [\mathbf{F}_1, \mathbf{F}_2, \mathbf{F}_3]$, $\mathbf{F}_k = [f_1, \ldots, f_p]^T$, $k = 1, 2, 3$, and random initial data

$$\mathbf{U}(\mathbf{x}, 0, \omega) = \mathbf{U}_0(\mathbf{x}, \omega), \ \omega \in \Omega. \tag{7.2}$$

A mathematical framework of *random entropy solutions* for scalar conservation laws has been developed in [16]. There, existence and uniqueness of random entropy solutions to (7.1)–(7.2) has been shown for scalar hyperbolic conservation laws, also in multiple dimensions. Furthermore, the existence of the statistical quantities of the random entropy solution such as the statistical mean and k-point spatio-temporal correlation functions under suitable assumptions on the random initial data have been proven.

7.3 Stochastic FVM

We parametrize all the random inputs in the Eqs. (7.1)–(7.2) using the random variable $\mathbf{y} = \mathbf{Y}(\omega)$ which takes values in $D_y \subset \mathbb{R}^q$ and rewrite the stochastic conservation law in the parametric form:

$$\partial_t \mathbf{U} + \nabla_x \cdot \mathbf{F}(\mathbf{U}, \mathbf{y}) = \mathbf{0},\ \mathbf{x} \in D_x \subset \mathbb{R}^3,\ \mathbf{y} \in D_y \subset \mathbb{R}^q,\ t > 0; \tag{7.3}$$

$$\mathbf{U}(\mathbf{x}, 0, \mathbf{y}) = \mathbf{U}_0(\mathbf{x}, \mathbf{y}),\ \mathbf{x} \in D_x \subset \mathbb{R}^3,\ \mathbf{y} \in D_y \subset \mathbb{R}^q. \tag{7.4}$$

Let $\mathscr{T}_X = \cup_{i=1}^{N_x} K_x^i$ be the triangulation of the computational domain D_x in the physical space and $\mathscr{C}_y = \cup_{j=1}^{N_y} K_y^j$ be the Cartesian grid in the domain D_y of the parametrized probability space.

We further assume the existence of the probability density function $\mu(\mathbf{y})$ and compute the expectation of the n-th solution component of the conservation law (7.3)–(7.4) as follows:

$$\mathbb{E}[u_n] = \int_{D_y} u_n \mu(\mathbf{y})\, d\mathbf{y},\ n = 1, \ldots, p$$

The scheme of the Stochastic Finite Volume method (SFVM) [1–3] can be obtained from the integral form of the Eqs. (7.3)–(7.4):

$$\iint\limits_{K_y^j\, K_x^i} \partial_t \mathbf{U}\, \mu(\mathbf{y})\, d\mathbf{x} d\mathbf{y} + \iint\limits_{K_y^j\, K_x^i} \nabla_x \cdot \mathbf{F}(\mathbf{U}, \mathbf{y})\, \mu(\mathbf{y})\, d\mathbf{x} d\mathbf{y} = \mathbf{0}.$$

Introducing the cell average

$$\bar{\mathbf{U}}_{ij}(t) = \frac{1}{|K_x^i||K_y^j|} \iint\limits_{K_y^j\, K_x^i} \mathbf{U}(\mathbf{x}, t, \mathbf{y}) \mu(\mathbf{y})\, d\mathbf{x} d\mathbf{y}$$

with the cell volumes

$$|K_x^i| = \int_{K_x^i} d\mathbf{x}, \quad |K_y^j| = \int_{K_y^j} \mu(\mathbf{y})\, d\mathbf{y}$$

and performing the partial integration over K_x^i we get

$$\frac{d\bar{\mathbf{U}}_{ij}}{dt} + \frac{1}{|K_x^i||K_y^j|} \int_{K_y^j} \left[\int_{K_x^i} \mathbf{F}(\mathbf{U}, \mathbf{y}) \cdot \mathbf{n}\, dS \right] \mu(\mathbf{y})\, d\mathbf{y} = \mathbf{0}$$

Next, we use any standard numerical flux approximation $\hat{\mathbf{F}}\big(\tilde{\mathbf{U}}_L(\mathbf{x}, t, \mathbf{y}), \tilde{\mathbf{U}}_R(\mathbf{x}, t, \mathbf{y}), \mathbf{y}\big)$ to replace the discontinuous flux through the element interface $\mathbf{F}(\mathbf{U}, \mathbf{y}) \cdot \mathbf{n}$.

Here $\tilde{\mathbf{U}}_{L,R}$ denote the boundary extrapolated solution values at the edge of the cell K_x^i, obtained by the high order reconstruction from the cell averages. The complete numerical flux is then approximated by a suitable quadrature rule as

$$\bar{\mathbf{F}}_{ij}(t) = \frac{1}{|K_y^j|} \int\limits_{K_y^j} \left[\int\limits_{K_x^i} \hat{\mathbf{F}}(\tilde{\mathbf{U}}_L, \tilde{\mathbf{U}}_R, \mathbf{y}) \right] \mu(\mathbf{y})\, d\mathbf{y} \approx \frac{1}{|K_y^j|} \sum_{\mathbf{m}} \hat{\mathscr{F}}(t, \mathbf{y}_\mathbf{m}) \mu(\mathbf{y}_\mathbf{m}) w_\mathbf{m}, \tag{7.5}$$

where we have denoted the flux integral over the physical cell as $\hat{\mathscr{F}}$, $\mathbf{m} = (m_1, \ldots, m_q)$ is the multi-index, $\mathbf{y}_\mathbf{m}$ and $w_\mathbf{m}$ are quadrature nodes and weights, respectively.

The SFV method then results in the solution of the following ODE system:

$$\frac{d\bar{\mathbf{U}}_{ij}}{dt} + \frac{1}{|K_x^i|} \bar{\mathbf{F}}_{ij}(t) = \mathbf{0}, \tag{7.6}$$

for all $i = 1, \ldots, N_x$, $j = 1, \ldots, N_y$. Therefore, to obtain the high-order scheme we first need to provide the high-order flux approximation based, for example, on the ENO/WENO reconstruction in the physical space. Second, we have to guarantee the high-order integration in (7.5) also by applying the ENO/WENO reconstruction in the stochastic space and choosing the suitable quadrature rule. Finally, we need the high-order time-stepping algorithm to solve the ODE system (7.6), such as Runge-Kutta method.

7.4 Convergence and Efficiency Analysis for SFVM

In this section we present the results of the convergence and efficiency analysis of the SFV method applied to the stochastic scalar conservation law in the parametric form

$$\partial_t u + \partial_x f(u; y) = 0, \quad x \in D,\ y \in Y,\ t > 0, \tag{7.7}$$

$$u(x, 0; y) = u_0(x; y). \tag{7.8}$$

Let $Y = \bigcup\limits_{k=1}^{N_y} Y_k$, where Y_k is the mesh cell in the stochastic variable y. Denote the probability density function by $\mu(y)$.

7.4.1 Convergence of Statistical Moments in L^1-Norm

Denote by u the exact solution of (7.7), by u_h^y the numerical solution which is exact in x variable and discretized in y and by u_h^{xy} the numerical discretized in both

variables. Assume that the numerical solution converges with rate p in x variable and rate r in y variable, that is

$$\|u_h^y - u_h^{xy}\|_{L^1(D)} \le C_1 \Delta x^p \ \forall y \in Y. \tag{7.9}$$

$$\|u - u_h^y\|_{L^1(Y)} \le C_2 \Delta y^r \ \forall x \in D, \tag{7.10}$$

The next estimate follows immediately from this assumption:

$$\|u - u_h^{xy}\|_{L^1(D\times Y)} \le C_1 \Delta x^p + C_2 \Delta y^r. \tag{7.11}$$

Convergence of $E_h[u_h^{xy}]$ in L^1-norm The expected value of the exact solution is a deterministic function

$$\mathbb{E}[u](x_i, t^n) = \int_Y u(x_i, t^n; y)\mu(y)\, dy, \tag{7.12}$$

and the approximation of the expectation of the numerical solution is, as before, equal to

$$E_h[u_h^{xy}]_i^n = \sum_{k=1}^{N_y} u_{ik}^n \omega_k = \sum_{k=1}^{N_y} u_{ik}^n \int_{Y_k} \mu(y)\, dy = \sum_{k=1}^{N_y} \int_{Y_k} u_{ik}^n\, \mu(y)\, dy = \mathbb{E}[u_h^{xy}](x_i, t^n). \tag{7.13}$$

Then

$$\begin{aligned}
\left\|\mathbb{E}[u] - \mathbb{E}[u_h^{xy}]\right\|_{L^1(D)} &= \left\|\mathbb{E}[u] - \mathbb{E}[u_h^y] + \mathbb{E}[u_h^y] - \mathbb{E}[u_h^{xy}]\right\|_{L^1(D)} \le \\
&\le \left\|\mathbb{E}[u] - \mathbb{E}[u_h^y]\right\|_{L^1(D)} + \left\|\mathbb{E}[u_h^y] - \mathbb{E}[u_h^{xy}]\right\|_{L^1(D)} = \\
&= \int_D \left|\mathbb{E}[u] - \mathbb{E}[u_h^y]\right| dx + \int_D \left|\mathbb{E}[u_h^y] - \mathbb{E}[u_h^{xy}]\right| dx = \\
&= \int_D \left| \int_Y (u - u_h^y)\mu(y)\, dy \right| dx + \int_D \left| \int_Y (u_h^y - u_h^{xy})\mu(y)\, dy \right| dx \le \\
&\le \int_D \int_Y |u - u_h^y| \mu(y)\, dy dx + \int_D \int_Y |u_h^y - u_h^{xy}| \mu(y)\, dy dx.
\end{aligned} \tag{7.14}$$

The first integral in (7.14) can be estimated as follows:

$$\int_D \int_Y |u - u_h^y| \mu(y)\, dy dx \le \int_D \sup_Y \mu(y) \int_Y |u - u_h^y|\, dy dx = C \|u - u_h^y\|_{L^1(Y)} \le C \Delta y^r, \tag{7.15}$$

and for the second integral we have

$$\int_D \int_Y |u_h^y - u_h^{xy}| \mu(y)\, dydx = \int_Y \Big[\int_D |u_h^y - u_h^{xy}|\, dx \Big] \mu(y)\, dy =$$

$$= \|u_h^y - u_h^{xy}\|_{L^1(D)} \int_Y \mu(y)\, dy = \|u_h^y - u_h^{xy}\|_{L^1(D)} \le C\Delta x^p. \quad (7.16)$$

Hence, the convergence rate of the expectation in L^1-norm can be estimated as

$$\big\|\mathbb{E}[u] - \mathbb{E}[u_h^{xy}]\big\|_{L^1(D)} \le C_1\Delta x^p + C_2\Delta y^r. \quad (7.17)$$

Convergence of $V_h[u_h^{xy}]$ in L^1-norm The variance of the exact solution at (x_i, t^n) is equal to

$$\mathbb{V}[u](x_i,t^n) = \mathbb{E}\big[\big(u(x_i,t^n) - \mathbb{E}[u](x_i,t^n)\big)^2\big] = \mathbb{E}\big[u^2(x_i,t^n)\big] - \big(\mathbb{E}[u](x_i,t^n)\big)^2, \quad (7.18)$$

and can be approximated as

$$V_h[u_h^{xy}]_i^n = E_h\big[(u_h^{xy})^2\big]_i^n - \big(E_h[u_h^{xy}]_i^n\big)^2 = \mathbb{E}\big[(u_h^{xy})^2\big]_i^n - \big(\mathbb{E}[u_h^{xy}]_i^n\big)^2 = \mathbb{V}[u_h^{xy}]_i^n. \quad (7.19)$$

Then

$$\big\|\mathbb{V}[u] - \mathbb{V}[u_h^{xy}]\big\|_{L^1(D)} = \big\|\mathbb{E}\big[u^2\big] - \big(\mathbb{E}[u]\big)^2 - \mathbb{E}\big[(u_h^{xy})^2\big] + \big(\mathbb{E}[u_h^{xy}]\big)^2\big\|_{L^1(D)} \le$$

$$\le \big\|\mathbb{E}\big[u^2\big] - \mathbb{E}\big[(u_h^{xy})^2\big]\big\|_{L^1(D)} + \big\|\big(\mathbb{E}[u]\big)^2 - \big(\mathbb{E}[u_h^{xy}]\big)^2\big\|_{L^1(D)}. \quad (7.20)$$

The following estimate holds for the first integral in (7.20):

$$\big\|\mathbb{E}\big[u^2\big] - \mathbb{E}\big[(u_h^{xy})^2\big]\big\|_{L^1(D)} = \int_D \big|\mathbb{E}\big[u^2\big] - \mathbb{E}\big[(u_h^{xy})^2\big]\big|\, dx =$$

$$= \int_D \Big| \int_Y \big[u^2 - (u_h^{xy})^2\big] \mu(y)\, dy \Big| dx \le \int_D \int_Y \big|u^2 - (u_h^{xy})^2\big| \mu(y)\, dydx =$$

$$= \int_D \int_Y \big|u - u_h^{xy}\big|\big|u + u_h^{xy}\big| \mu(y)\, dydx \le C \int_D \int_Y \big|u - u_h^{xy}\big| dydx =$$

$$= C\,\|u - u_h^{xy}\|_{L^1(D\times Y)} \le C\Delta x^p + C_7\Delta y^r. \quad (7.21)$$

For the second integral in (7.20) we get

$$\left\| \left(\mathbb{E}[u]\right)^2 - \left(\mathbb{E}[u_h^{xy}]\right)^2 \right\|_{L^1(D)} = \int_D \left| \left(\mathbb{E}[u]\right)^2 - \left(\mathbb{E}[u_h^{xy}]\right)^2 \right| dx =$$

$$= \int_D \left| \mathbb{E}[u] - \mathbb{E}[u_h^{xy}] \right| \left| \mathbb{E}[u] + \mathbb{E}[u_h^{xy}] \right| dx \leq$$

$$\leq C \left\| \mathbb{E}[u] - \mathbb{E}[u_h^{xy}] \right\|_{L^1(D)} \leq C_1 \Delta x^p + C_2 \Delta y^r. \quad (7.22)$$

Finally, from (7.21) to (7.22) we get

$$\left\| \mathbb{V}[u] - \mathbb{V}[u_h^{xy}] \right\|_{L^1(D)} \leq C_1 \Delta x^p + C_2 \Delta y^r. \quad (7.23)$$

Similar estimates are also valid for higher moments of u.

7.4.2 *Efficiency Estimates*

In the previous section it has been shown that the error of the expectation approximation is given by

$$E = \left\| \mathbb{E}[u] - \mathbb{E}[u_h^{xy}] \right\|_{L^1(D)} \leq C_1 \Delta x^p + C_2 \Delta y^r, \quad (7.24)$$

where p and r are the convergence rates of the SFVM solver in physical and stochastic variables, respectively. Based on this result, we derive the error vs. work estimates for SFVM.

Let $x \in \mathbb{R}^n$, $y \in \mathbb{R}^m$. Assume that the CFL condition is satisfied, such that $\Delta t = O(\delta x)$. The total work W (or total time) required to compute the solution of the stochastic scalar conservation law using SFVM is proportional to the total numbers of grid points in x, y and t axes, denoted respectively by N_x, N_y and N_t, i.e.

$$W = C N_x N_y N_t = C \frac{1}{\Delta x^n} \frac{1}{\Delta y^m} \frac{1}{\Delta t} = \frac{C}{\Delta x^{n+1} \Delta y^m} = C \Delta x^{-(n+1)} \Delta y^{-m}. \quad (7.25)$$

Further derivation of the estimate depends on the choice of the mesh sizes equilibration, that is, on the relation between Δx and Δy.

1. Assume that the mesh sizes are equilibrated according to the expected orders of convergence p and r: $\Delta y = \Delta x^{p/r}$. Then $E = C \Delta x^p$ and $\Delta x = C E^{1/p}$. Substituting these relations into the Eq. (7.25) we get

$$W = C\Delta x^{-(n+1)}\Delta x^{-pm/r} = C\Delta x^{-(n+1+pm/r)} = CE^{-\frac{n+1+pm/r}{p}} \tag{7.26}$$

and hence

$$E = CW^{-\frac{p}{n+1+pm/r}}, \tag{7.27}$$

which is the desired error vs. work estimated.

2. Assume now that the mesh size Δy is obtained by the following scaling: $\Delta y = \eta\Delta x$, where η is the constant scaling factor, meaning that the stochastic mesh is isotropic (same Δy for all random variables). Define $q = \min(p, r)$. Then $E = C\Delta x^q$ and $\Delta x = CE^{1/q}$, and the total work is defined as

$$W = C\Delta x^{-(n+1)}\Delta y^{-m} = C\Delta x^{-(n+m+1)} = CE^{-\frac{n+m+1}{q}}, \tag{7.28}$$

which finally gives

$$E = CW^{-\frac{q}{n+m+1}}. \tag{7.29}$$

Note that the estimate (7.29) is equivalent to the complexity result for the deterministic finite-volume method in the $(n + m)$-dimensional space, which sets strict limitations on the number of random variables that can be handled by the SFVM if the scaling factor η is close to 1. However, computational practice shows that it is sufficient to use only few computational cells to discretize the equations in the stochastic space to obtain a good quality approximation of the statistical quantities and therefore the SFVM is essentially much more efficient as deterministic FVM. Another significant simplification of the approach is the absence of the fluxes in the stochastic variables y, which also contributes to the efficiency of the SFVM.

3. Assume that the stochastic mesh is anisotropic, that is the mesh sizes Δy_k are different for $k = 1, \ldots, m$: $\Delta y_k = \eta_k \Delta x$. Applying the same technique as above we obtain

$$E = C_1\Delta x + C_2\sum_{k=1}^{m}\Delta y_k^r = C_1\Delta x + C_2\Delta x^r\sum_{k=1}^{m}\eta_k^r \le C\Delta x^q\Big(1 + \sum_{k=1}^{m}\eta_k^r\Big), \tag{7.30}$$

where $q = \min(p, r)$ as before. We have also assumed that $\Delta x << 1$ such that $\Delta x^p < \Delta x^q$ and $\Delta x^r < \Delta x^q$. Then the mesh size Δx can be represented as

$$\Delta x = \left(\frac{E}{1 + \sum\limits_{k=1}^{m}\eta_k^r}\right)^{1/q}. \tag{7.31}$$

The total work is

$$W = C\Big(\prod_{k=1}^{m}\eta_k^{-1}\Big)\Delta x^{-(n+m+1)} = C\Big(\prod_{k=1}^{m}\eta_k^{-1}\Big)\left(\frac{E}{1+\sum\limits_{k=1}^{m}\eta_k^r}\right)^{-(n+m+1)/q}, \tag{7.32}$$

and the resulting error vs. work estimate is

$$E = C\big(1+\sum_{k=1}^{m}\eta_k^r\big)\Big(\prod_{k=1}^{m}\eta_k^{-\frac{q}{n+m+1}}\Big)W^{-\frac{q}{n+m+1}}. \tag{7.33}$$

Note that in the isotropic case, when all $\eta_k = \eta = \text{const}$, formula (7.33) results in

$$E = C(1+m\eta^r)\,\eta^{-\frac{qm}{n+m+1}}\,W^{-\frac{q}{n+m+1}}. \tag{7.34}$$

Comparing (7.33) and (7.34) we notice that the proper choice of scaling factors η_k in the anisotropic stochastic mesh construction, while not affecting the convergence *rates*, can reduce the convergence constant, which means increasing computational efficiency. The choice of η_k should be based on the sensitivity analysis of the random entropy solution to each of the m random variables.

Let us demonstrate the efficiency provided by the anisotropic mesh adaptation. We compare the convergence constants:

$$C_i = (1+m\eta^r)\,\eta^{-\frac{qm}{n+m+1}} \tag{7.35}$$

for the *isotropic* mesh with equal mesh sizes in all stochastic coordinates, $\Delta y_k = \eta\Delta x$, $k = 1,\ldots,m$, and

$$C_a = \big(1+\sum_{k=1}^{m}\eta_k^r\big)\prod_{k=1}^{m}\eta_k^{-\frac{q}{n+m+1}} \tag{7.36}$$

for the *anisotropic* stochastic mesh with mesh size scaling according to $\Delta y_k = \eta_k\Delta x$, $k = 1,\ldots,m$. Assume further that $\eta_1 > \eta_2 > \ldots \eta_m$ and $\eta_k > 1$ for all k, such that $\Delta y_k > \Delta x$.

Our goal is to show that the convergence constants ratio $\delta_m = \dfrac{C_i}{C_a} > 1$ as $m \to \infty$ if $\eta_1 < \eta$ and $r > q$, that is, the anisotropic stochastic mesh increases the algorithm efficiency as the number of random variables grows if the convergence rate r in the stochastic space is higher than q, the minimum of the convergence rates in physical and stochastic coordinates.

We start by noting that under the assumption $\eta_1 < \eta$ the following inequality is valid:

$$C_a = \left(1 + \sum_{k=1}^{m} \eta_k^r\right) \prod_{k=1}^{m} \eta_k^{-\frac{q}{n+m+1}} < (1 + m\eta_1^r)\eta_1^{-\frac{qm}{n+m+1}} = C_a^1, \tag{7.37}$$

and therefore

$$\delta_m = \frac{C_i}{C_a} > \frac{C_i}{C_a^1} = \frac{(1 + m\eta^r)\,\eta^{-\frac{qm}{n+m+1}}}{(1 + m\eta_1^r)\,\eta_1^{-\frac{qm}{n+m+1}}} = \left(\frac{1 + m\eta^r}{1 + m\eta_1^r}\right)\left(\frac{\eta}{\eta_1}\right)^{-\frac{qm}{n+m+1}}. \tag{7.38}$$

Hence, the limit of the constants ratio is

$$\delta = \lim_{m\to\infty} \delta_m = \left(\frac{\eta}{\eta_1}\right)^{r-q}, \tag{7.39}$$

and clearly $\delta > 1$ if $r > q$.

Let's analyse in more detail the possible values of δ in dependence on the convergence rates p and r in x and y variables, respectively.

Smooth solution. If the solution is smooth in x and y, then the convergence rate of the SFVM is the expected one, therefore by applying high-order finite-volume approximations in both variables one can obtain the full convergence rates p and r.

- If $p < r$, then $q = \min(p, r) = p$ and $r - q = r - p > 0$, $\delta > 1$ and hence the SFVM will converge faster on anisotropic stochastic mesh.
- If $p > r$, then $q = \min(p, r) = p$ and $r - q = r - p < 0$, $\delta < 1$, therefore the anisotropic mesh doesn't improve the convergence.

Shock solution. Recall that if the shock wave appears in the physical space, then it also propagates into the stochastic space, so that the solution becomes discontinuous in both x and y. In this case the 1st order FV scheme one typically gives $p = 1/2$ according to and $r = 1$ as shown in [18], therefore $q = \min(p, r) = 1/2$ and $\delta = \sqrt{\frac{\eta}{\eta_1}} > 1$. This means that the SFVM on anisotropic mesh in the stochastic space is more efficient than SFVM on the uniform mesh even if the solution has a shock.

7.4.3 Numerical Convergence Analysis

We perform the convergence analysis of the SFVM for a simple linear advection equation with uncertain phase initial condition

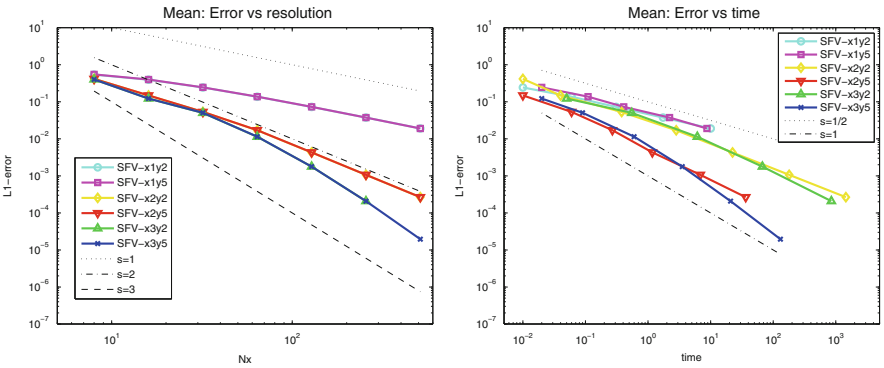


Fig. 7.1 Mean: dependence of the error on the mesh resolution and computational time

$$u_t + a u_x = 0, \quad x \in (0,1),$$
$$u(x,0) = \sin\left(2\pi\left(x + 0.1Y(\omega)\right)\right).$$

The random variable $y = Y(\omega)$ is assumed to be distributed uniformly on $[0,1]$.

In Figs. 7.1–7.2, we plot the $L^1(0,1)$ error for the expectation and the variance of u with respect to the mesh size and the computational time. We investigate the influence of different reconstruction orders in spacial and stochastic variables on the convergence rates and therefore present the convergence plots for the SFVM based on different combinations of ENO/WENO reconstruction in x and y. We compare the SFVM with 1st, 2nd and 3rd order of accuracy in physical space combined with 3rd and 5th order reconstruction in stochastic variable. The type of reconstruction is indicated in Figs. 7.1–7.2 as follows: for example, the line "SFV-x2y5" corresponds to the 2nd order piecewise-linear ENO reconstruction in x and 5th order piecewise-quadratic WENO reconstruction in y, the line "SFV-x3y5" stands for 3rd order piecewise-linear WENO reconstruction in x with 5th order WENO reconstruction in y, etc. The numerical flux used in all the numerical experiments of this paper is the Rusanov flux. The results show that, while the convergence rate is dominated by the order of accuracy in x, the algorithms with higher order reconstruction in y are more efficient computationally since the same error can be reached with less overall computational time as compared to the lower order reconstruction in y.

7.4.4 Numerical Results

7.4.4.1 Stochastic Sod's Shock Tube Problem

Consider the Riemann problem for the one-dimensional Euler equations with randomness in both flux and initial data

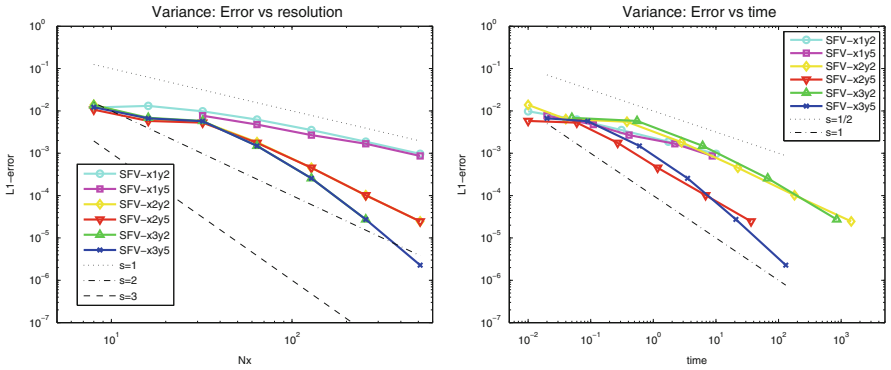


Fig. 7.2 Variance: dependence of the error on the mesh resolution and computational time

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{F}(\mathbf{U},\omega)}{\partial x} = \mathbf{0}, \quad x \in (0,2), \tag{7.40}$$

$$\mathbf{U}(x,0,\omega) = \mathbf{U}_0\big(x, Y_1(\omega), Y_2(\omega)\big) = \begin{cases} \mathbf{U}_L\big(Y_2(\omega)\big), & x < Y_1(\omega); \\ \mathbf{U}_R, & x > Y_1(\omega), \end{cases} \tag{7.41}$$

with $y_j = Y_j(\omega),\ j = 1,2,3,\ \omega \in \Omega$ and

$$\mathbf{U} = [\rho, \rho u, E]^T,\ \mathbf{F} = [\rho u, \rho u^2 + p, \rho u(E+p)]^T,$$

$$p = (\gamma - 1)\Big(E - \frac{1}{2}\rho u^2\Big).$$

We also assume the randomness in the adiabatic constant, $\gamma = \gamma\big(Y_3(\omega)\big)$, and therefore

$$\mathbf{F}(\mathbf{U},\omega) = \mathbf{F}\big(\mathbf{U}, Y_3(\omega)\big).$$

The initial data is set in primitive variables as

$$\mathbf{W}_0(x,\omega) = [\rho_0, u_0, p_0]^T = \begin{cases} [1.0, 0.0, 1.0] & \text{if} \quad x < Y_1(\omega), \\ [0.125 + 0.5\ Y_2, 0.0, 0.1] & \text{if} \quad x > Y_1(\omega). \end{cases}$$

We apply the SFVM to solve the system (7.40)–(7.41) with $Y_1(\omega) \sim \mathscr{U}[0.95, 1.05]$, $Y_2(\omega) \sim \mathscr{U}[-0.1, 0.1]$, $Y_3(\omega) \sim \mathscr{U}[1.2, 1.6]$ using the 3rd order WENO reconstruction in both physical and stochastic variables. The results are presented in Figs. 7.3–7.4, in which the solution mean (solid line) as well as mean plus/minus standard deviation (dashed lines) are plotted.

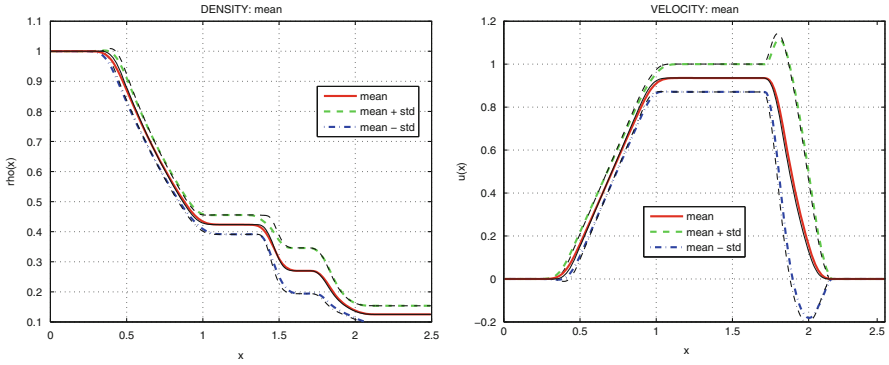


Fig. 7.3 Sod's shock tube problem with random flux and initial data: density (*left*) and velocity (*right*)

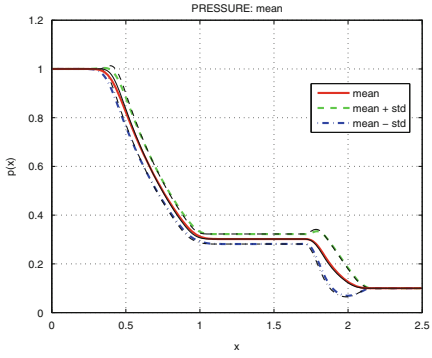


Fig. 7.4 Sod's shock tube problem with random flux and initial data: pressure

The convergence results (dependence of the error on the number of mesh points) for the solution mean are presented in Fig. 7.5. Due to the shock formation in the path-wise solution the maximum order of convergence for the mean is limited to 1st.

We compare the efficiencies of the SFV and MLMC methods [16, 17] for the solution of the one-dimensional stochastic Sod's problem for the Euler equations. Figure 7.6 illustrates the convergence of SFVM and MLMC based on 1st and 2nd order FV ENO/WENO solvers.

Figure 7.6 demonstrates that both approaches lead to the same orders of convergence in space while SFVM with properly chosen reconstruction orders appears to be more efficient in terms of error-to-work estimates.

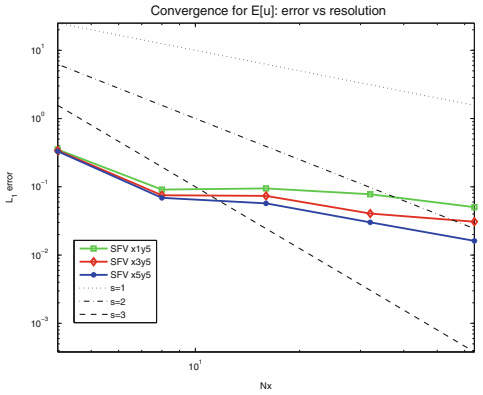


Fig. 7.5 Sod's shock tube problem with random flux and initial data: convergence

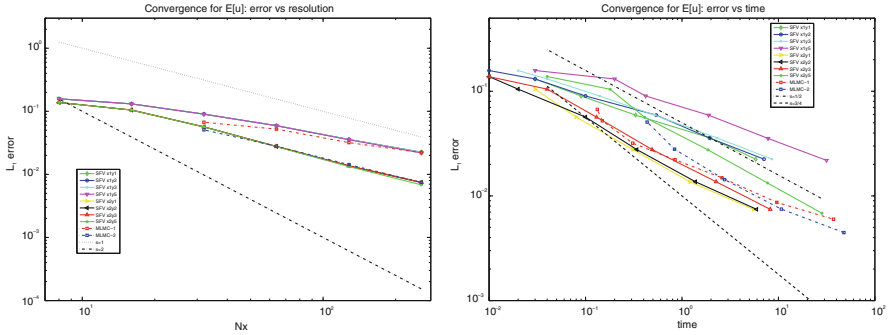


Fig. 7.6 Convergence of the SFVM and MLMC

7.4.4.2 Stochastic Cloud-Shock Interation (Random IC)

Consider the two-dimensional Euler equations with random initial data:

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{F}(\mathbf{U})}{\partial x_1} + \frac{\partial \mathbf{G}(\mathbf{U})}{\partial x_2} = \mathbf{0}, \quad (x_1, x_2) \in [0,1] \times [0,1], \tag{7.42}$$

$$\mathbf{U}(x, 0, \omega) = \mathbf{U}_0\big(x, Y(\omega)\big), \tag{7.43}$$

where

$$\mathbf{U} = [\rho, \rho u, \rho v, E]^T,$$

$$\mathbf{F} = [\rho u, \rho u^2 + p, \rho uv, \rho u(E + p)]^T,$$

$$\mathbf{G} = [\rho u, \rho uv, \rho v^2 + p, \rho v(E + p)]^T,$$

$$p = (\gamma - 1)\Big(E - \frac{1}{2}\rho(u^2 + v^2)\Big).$$

We apply the SFV method to solve the stochastic shock-vortex interaction problem for the Euler equations (see [21] for the deterministic version of the problem). The initial Mach 1.1 shock wave is normal to the x_1-axis and has uncertain location at $x_1 = 0.5 + 0.1Y(\omega)$ with $Y(\omega) \sim \mathscr{U}[0, 1]$. The states in front and behind the shock wave are:

$$[\rho_0, u_0, v_0, p_0] = \begin{cases} [1, \sqrt{\gamma}, 0, 1], & \text{if } x_1 < Y(\omega), \\ [\frac{1}{1.1}, 1.1\sqrt{\gamma}, 0, 1 - \frac{\gamma}{10}], & \text{if } x_1 > Y(\omega). \end{cases}$$

A small vortex centered at $(x_1^c, x_2^c) = (0.25, 0.5)$ is superposed to the flow left to she shock and is described as a perturbation to the velocity (u, v) and pressure p:

$$\tilde{u}_0 = \epsilon\tau e^{\alpha(1-\tau^2)} \sin\theta,$$

$$\tilde{v}_0 = -\epsilon\tau e^{\alpha(1-\tau^2)} \cos\theta,$$

$$\tilde{p}_0 = -(\gamma - 1)\frac{\epsilon^2 e^{2\alpha(1-\tau^2)}}{4\alpha\gamma}\rho_0,$$

where

$$r = \sqrt{(x_1 - x_1^c)^2 + (x_2 - x_2^c)^2},\ \tau = \frac{r}{r_c},\ \sin\theta = \frac{x_2 - x_2^c}{r},\ \cos\theta = \frac{x_1 - x_1^c}{r}.$$

Here ϵ indicates the vortex strength, α controls the decay rate of the vortex and r_c is the critical radius for which the vortex has the maximum strength. In our test we choose $\epsilon = 0.3$, $r_c = 0.05$ and $\alpha = 0.204$.

For this computation, we have used the SFV method based on 5th order WENO solver in the physical space and 2nd order ENO solver in the stochastic space on the uniform 128×128 cells Cartesian grid in the physical space and 32 cells grid in the stochastic coordinate. The computational results for the flow mean and variance at $t = 0.35$ are presented in Figs. 7.7–7.10.

7.5 Stochastic DG/FVM

7.5.1 Stochastic DG/FV Method

In this section we generalize the approach to uncertainty quantification described previously in order to efficiently apply high-order approximation techniques on

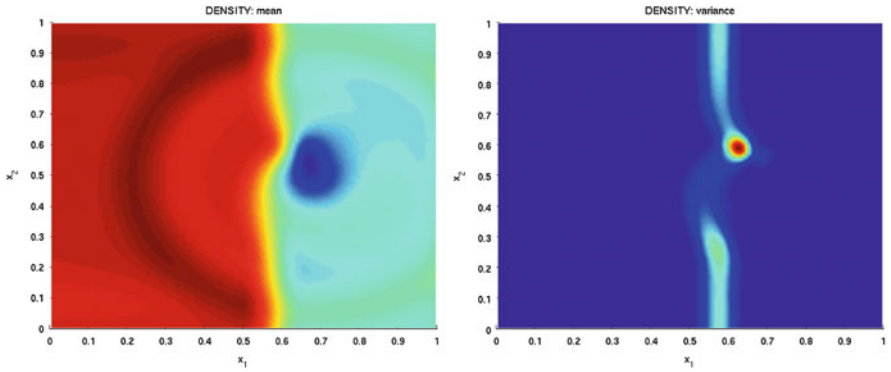


Fig. 7.7 Density: mean (*left*) and variance (*right*) at $t = 0.35$

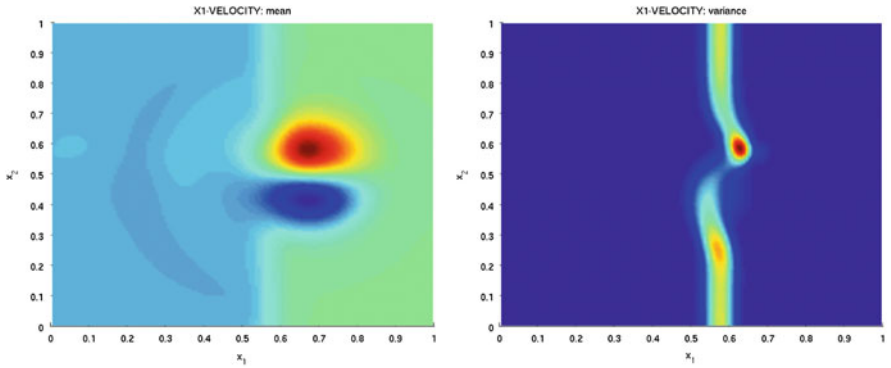


Fig. 7.8 x_1-velocity: mean (*left*) and variance (*right*) at $t = 0.35$

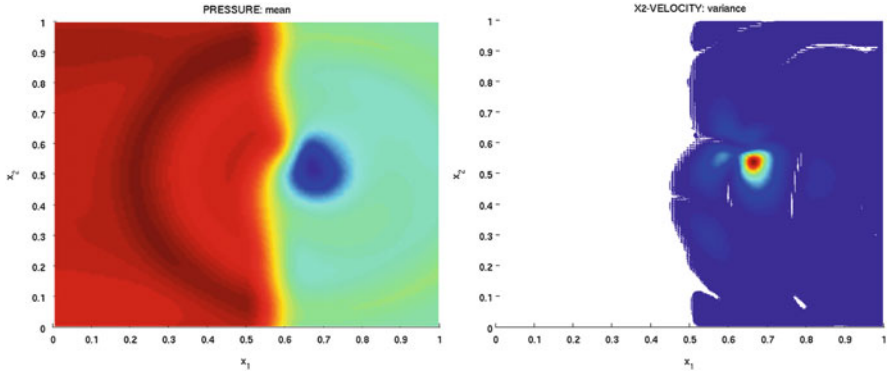


Fig. 7.9 x_2-velocity: mean (*left*) and variance (*right*) at $t = 0.35$

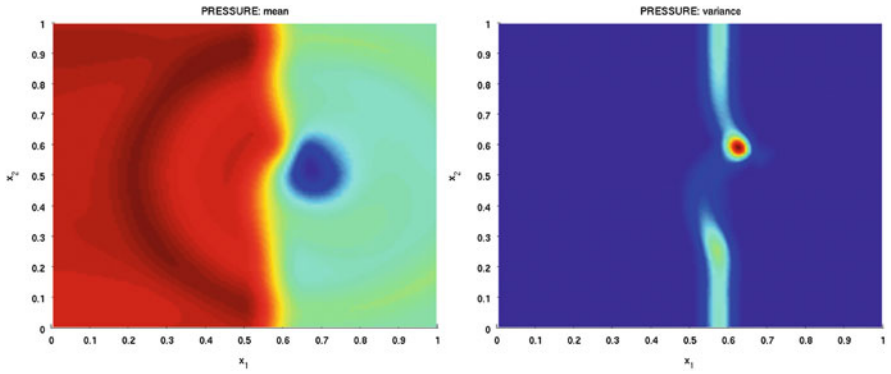


Fig. 7.10 Pressure: mean (*left*) and variance (*right*) at $t = 0.35$

unstructured grid in physical domains with complicated geometry. To this end, we use the Discontinuous Galerkin (DG) method [4] to discretize the equations in the physical space and combine it with the finite-volume discretization in the stochastic variables as described in Sect. 7.3. Note that we can still use Cartesian grids in the stochastic space since the computational domain in this space is a q-dimensional rectangle.

As before, we start with the parametric form of the stochastic conservation law:

$$\partial_t \mathbf{U} + \nabla_x \cdot \mathbf{F}(\mathbf{U}, \mathbf{y}) = \mathbf{0}, \ \mathbf{x} \in D_x \ \subset \mathbb{R}^3, \ \mathbf{y} \in D_y \subset \mathbb{R}^q, \ t > 0; \tag{7.44}$$

$$\mathbf{U}(\mathbf{x}, 0, \mathbf{y}) = \mathbf{U}_0(\mathbf{x}, \mathbf{y}), \ \mathbf{x} \in D_x \ \subset \mathbb{R}^3, \ \mathbf{y} \in D_y \subset \mathbb{R}^q. \tag{7.45}$$

On each element K_x^i of the physical domain triangulation we apply the DG solution discretization in the physical variable $\mathbf{x}$:

$$\mathbf{U}_h(\mathbf{x}, t, \mathbf{y}) = \sum_{l=1}^{p} \mathbf{U}_l(t, \mathbf{y}) \varphi_l(\mathbf{x}), \ x \in K_x^i, \tag{7.46}$$

which leads to the following semi-discrete DG formulation:

$$\sum_{l=1}^{p} \partial_t \mathbf{U}_l(t, \mathbf{y}) \int_{K_x^i} \varphi_l(\mathbf{x}) \varphi_k(\mathbf{x}) \, d\mathbf{x} + \int_{\partial K_x^i} \mathbf{F}(\mathbf{U}_h, \mathbf{y}) \cdot \mathbf{n} \, \varphi_k(\mathbf{x}) \, d\mathbf{x} - \\ - \int_{K_x^i} \mathbf{F}(\mathbf{U}_h, \mathbf{y}) \nabla \varphi_k(\mathbf{x}) \, d\mathbf{x} = \mathbf{0}. \tag{7.47}$$

Note that at this stage the DG coefficients are still functions of the random variable $\mathbf{y}$ and time t and thus to get rid of this dependence we apply the finite-volume discretization in the random variable, which leads to

$$\sum_{l=1}^{p} \int\limits_{K_y^j} \left[\partial_t \mathbf{U}_l(t,\mathbf{y}) \int\limits_{K_x^i} \varphi_l(\mathbf{x})\varphi_k(\mathbf{x})\, d\mathbf{x} \right] \mu(\mathbf{y})\, d\mathbf{y} + \\ + \int\limits_{K_y^j} \left[\int\limits_{\partial K_x^i} \mathbf{F}(\mathbf{U}_h,\mathbf{y}) \cdot \mathbf{n}\, \varphi_k(\mathbf{x})\, d\mathbf{x} \right] \mu(\mathbf{y})\, d\mathbf{y} - \\ - \int\limits_{K_y^j} \left[\int\limits_{K_x^i} \mathbf{F}(\mathbf{U}_h,\mathbf{y}) \nabla \varphi_k(\mathbf{x})\, d\mathbf{x} \right] \mu(\mathbf{y})\, d\mathbf{y} = \mathbf{0}. \quad (7.48)$$

Finally, the resulting scheme becomes

$$\sum_{l=1}^{p} \frac{d\mathbf{U}_{lk}(t)}{dt} \int\limits_{K_x^i} \varphi_l(\mathbf{x})\varphi_k(\mathbf{x})\, d\mathbf{x} + \frac{1}{|K_y^j|} \iint\limits_{K_y^j \partial K_x^i} \mathbf{F}(\mathbf{U}_h,\mathbf{y}) \cdot \mathbf{n}\, \varphi_k(\mathbf{x})\, \mu(\mathbf{y}) d\mathbf{x} d\mathbf{y} - \\ - \frac{1}{|K_y^j|} \iint\limits_{K_y^j K_x^i} \mathbf{F}(\mathbf{U}_h,\mathbf{y}) \nabla \varphi_k(\mathbf{x})\, \mu(\mathbf{y}) d\mathbf{x} d\mathbf{y} = \mathbf{0}, \quad (7.49)$$

which is an ODE system with respect to the DG coefficients averaged over an element of the stochastic grid.

7.5.2 Numerical Results

7.5.2.1 Stochastic Cloud-Shock Interaction Problem (Random Flux)

Consider the two-dimensional Euler equations with deterministic initial data

$$[\rho_0, u_0, v_0, p_0] = \begin{cases} [3.86859, 11.2536, 0, 167.345], \text{ if } x_1 < 0.05, \\ [1, 0, 0, 1], \text{ if } x_1 > 0.05, \end{cases}$$

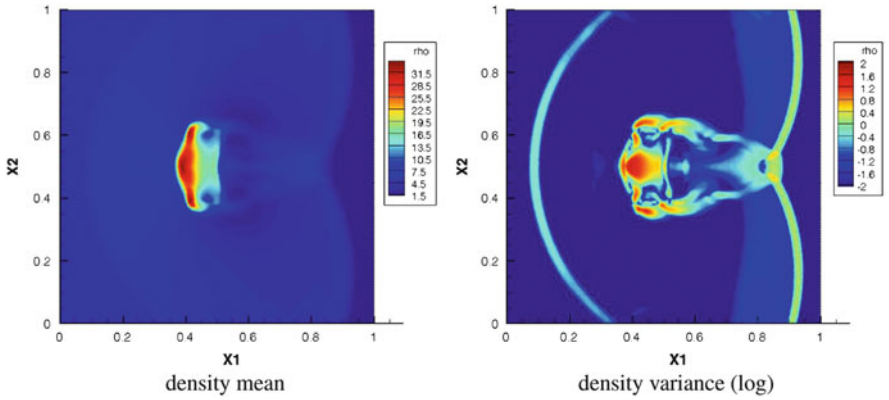


Fig. 7.11 Stochastic cloud-shock interaction problem

and a high-density cloud lying to the right of the shock:

$$\rho_0 = 10, \text{ if } \sqrt{(x_1 - 0.25)^2 + (x_2 - 0.5)^2} \leq 0.15.$$

Assume the random $\gamma = \gamma(\omega)$ in the equation of state (EOS)

$$p = (\gamma(\omega) - 1)\Big(E - \frac{1}{2}\rho(u^2 + v^2)\Big),$$

$$\gamma(\omega) \sim \mathscr{U}\big(5/3 - \epsilon, 5/3 + \epsilon\big),\ \epsilon = 0.1$$

The results of the simulation are presented in Fig. 7.11. In our computations we have used the 2nd order DG method in **x** variable and 3rd order WENO method in **y** variable, triangular mesh in **x** consisting of about 170,000 cells and Catresian mesh in **y** consisting of 16 cells. The results are plotted at $T = 0.06$.

7.5.2.2 Forward-Facing Step Channel

Consider the stochastic flow in the channel with the forward facing step with random Mach number of the inflowing gas: $\mathrm{M} \sim \mathscr{U}(2.9, 3.1)$. We have used the mesh of about 13,000 triangular cells in the physical space and 15 equally-sized cells in the stochastic space, the methods used are 2nd order DG and 3rd order WENO in physical and random variables, respectively. The results of the simulation are given in Fig. 7.12, indicating that the uncertainty in the Mach number influences the position and intensity of shock in front of the step, while having little effect on the shocks reflected from the channel walls.

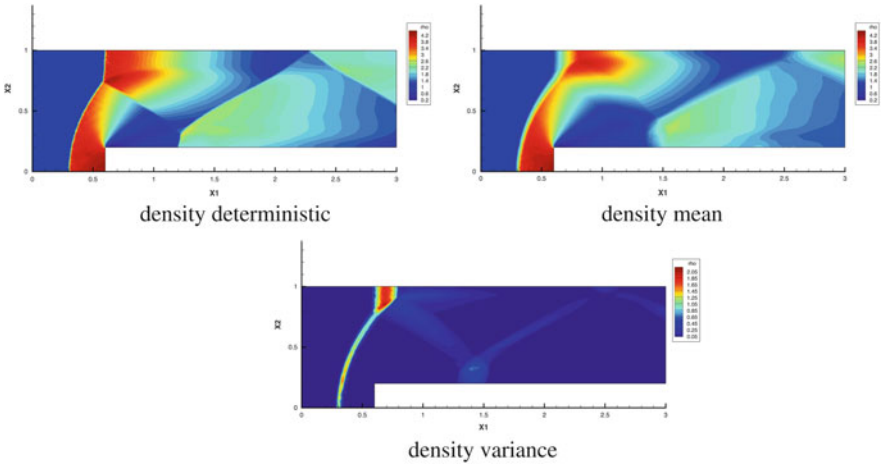


Fig. 7.12 Stochastic flow in a forward-facing step channel

7.6 Stochastic Mesh Adaptation

7.6.1 Stochastic Mesh Adaptation Based on Karhunen-Loève Expansion

In this section we discuss one mesh adaptation technique which can be used to reduce the computational cost of the SFV method. To this end, we consider the following model problem:

$$\frac{\partial u}{\partial t} + \frac{\partial f(u,\omega)}{\partial x} = 0,\ x \in D = [0, L] \subset \mathbb{R},\ t > 0; \tag{7.50}$$

$$u(x, 0, \omega) = u_0(x, \omega),\ x \in D,\ \omega \in \Omega. \tag{7.51}$$

Assume the following Karhunen-Loève expansion of the flux:

$$f(u;\omega) = \bar{f}(u) + \sum_{j \geq 1} Y_j(\omega)\sqrt{\lambda_j}\Phi_j(u), \tag{7.52}$$

where $\Phi_j(u)$ and λ_j are the eigenfunctions and eigenvalues of the integral operator with covariance kernel:

$$\int_D C_Y(u_1, u_2)\Phi(u_1)\,du_1 = \lambda\Phi(u_2).$$

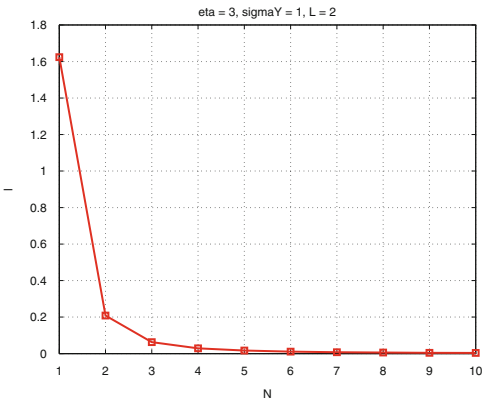


Fig. 7.13 Eigenvalues

We can therefore choose the random variable to parametrize the stochastic conservation law as $\boldsymbol{y} = (y_1, y_2, \dots) = \mathbf{Y}(\omega) = \big(Y_1(\omega), Y_2(\omega), \dots\big)$, then

$$f(u;\omega) = f(u;\boldsymbol{y})\Big|_{\boldsymbol{y}=\mathbf{Y}(\omega)} = \frac{u^2}{2} + \delta\Big(\sum_{j=1}^{q} y_j \sqrt{\lambda_j}\,\Phi_j(u)\Big).$$

Let $f(u;\omega)$ be the Gaussian process with exponential covariance [6]

$$C_Y(u_1,u_2) = \sigma_Y^2 e^{-|u_1-u_2|/\eta}, \quad \text{then}$$

$$\lambda_j = \frac{2\eta\sigma_Y^2}{\eta^2 w_j^2 + 1},\ \Phi_j(u) = \frac{1}{\sqrt{(\eta^2 w_j^2+1)L/2 + \eta}}[\eta w_j \cos(w_j u) + \sin(w_j u)],$$

where w_j are the roots of

$$(\eta^2 w^2 - 1)\sin(wL) = 2\eta w \cos(wL)$$

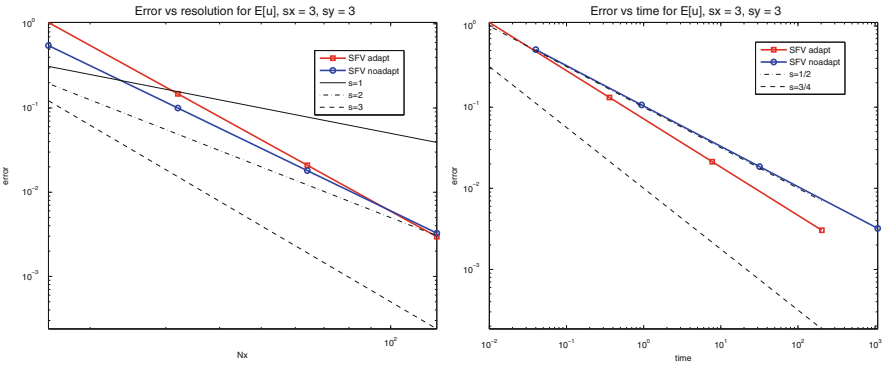
and

$$Y_j \sim \mathcal{N}(0,1), \quad \mathbb{E}[Y_j\ Y_k] = \delta_{jk}$$

Note that in this case the coefficients λ_j decay quickly in j (see Fig. 7.13).

Therefore the KL expansion can be truncated at moderate number of terms ($q = 2, 3$) without losing too much information about the stochastic process, namely

$$f(u;\boldsymbol{y}) = \bar{f}(u) + \sum_{j=1}^{q} y_j \sqrt{\lambda_j}\,\Phi_j(u)$$



error vs resolution error vs time

Fig. 7.14 Convergence of the SFVM: adaptive (*squares*) va. non-adaptive (*circles*) mesh

Consider the stochastic SCL with random flux and deterministic initial data

$$u_t + f(u;\omega)_x = 0,\ x \in (0, L),\ t > 0, \tag{7.53}$$

$$u(x, 0;\omega) = u_0(x) = 1 + \sin(\pi x). \tag{7.54}$$

In this paper, in order to reduce the computational cost of the SFV method, we propose the mesh adaptation in the stochastic space based on the choice of the number of nodes in each of the stochastic coordinates according to

$$N_y^j = C\ N_x \sqrt{\lambda_j}. \tag{7.55}$$

Figure 7.14 shows the convergence of the adaptive SFVM algorithm ("SFV adapt") and the SFVM without stochastic mesh adaptation ("SFV noadapt"). The non-adaptive version of the SFVM simply uses equal number of cells in each stochastic coordinate, while the adaptive version chooses the number of cells in each y_j according to (7.55). The computational time needed to perform both algorithms is shown in Fig. 7.15. Clearly, the proposed adaptation of the algorithm improves the convergence properties of the SFV method.

7.6.2 Numerical Results

7.6.2.1 Stochastic Cloud-Shock Interaction Problem (Random IC)

We use the mesh adaptation approach similar to the one described above to solve the stochastic cloud-shock interaction problem with initial data depending on *four* random variables. Note that the usage of non-adaptive algorithm for such simulation would lead to excessive computational cost of SFVM.

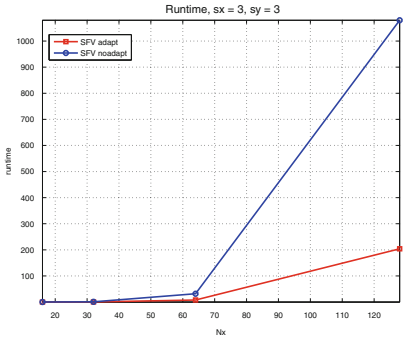


Fig. 7.15 Computational time: adaptive (*squares*) va. non-adaptive (*circles*) mesh

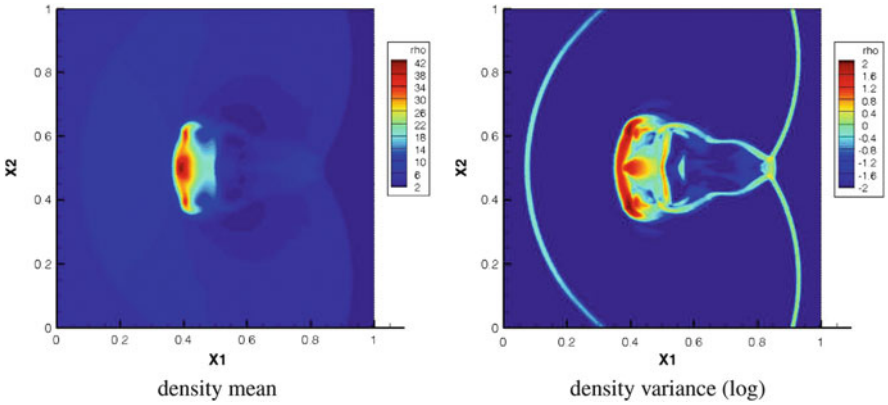


Fig. 7.16 Stochastic cloud-shock interaction problem

Consider the two-dimensional Euler equations with deterministic initial data

$$\mathbf{W}_0 = \begin{cases} [3.86859 + 0.1Y_2(\omega), 11.2536, 0, 167.345], \text{ if } x_1 < 0.04 + 0.01Y_1(\omega), \\ [1, 0, 0, 1], \text{ if } x_1 > 0.04 + 0.01Y_1(\omega), \end{cases}$$

with a high-density cloud to the right of the shock:

$$\rho_0 = 10 + 0.5Y_3(\omega), \text{ if } \sqrt{(x_1 - 0.25)^2 + (x_2 - 0.5)^2} \le 0.15 + 0.02Y_4(\omega).$$

The equations are closed by the following deterministic EOS: $p = (\gamma - 1)\Big(E - \frac{1}{2}\rho(u^2 + v^2)\Big)$, $\gamma = 5/3$. The random variables in the initial condition are uniformly distributed on $[0, 1]$: $Y_k \sim \mathscr{U}[0, 1]$, $k = 1, \ldots, 4$.

We use the 2nd order DG in $\mathbf{x}$ variable and 3rd order WENO in $\mathbf{y}$ variable, triangular mesh in $\mathbf{x}$ (170,000 cells) and *adaptive* Cartesian mesh in $\mathbf{y}$ ($3 \cdot 2 \cdot 7 \cdot 11 = 462$ cells), the output time is $T = 0.06$. The results of this simulation are illustrated in Fig. 7.16.

7.7 Conclusions

The SFV and SDG/FV methods studied in this paper appear to be a flexible and effective approach to the solution of stochastic conservation laws. We have shown that the SFV method it is applicable for the uncertainty quantification in a variety of complex problems including systems of conservation laws with random flux coefficients and initial data with several sources of uncertainty. The proper adaptation of the stochastic grid significantly reduces the computational cost of the method and improves its convergence.

References

1. Abgrall, R.: A simple, flexible and generic deterministic approach to uncertainty quantification in non-linear problems. Rapport de Recherce, INRIA, 00325315 (2007)
2. Abgrall, R., Congedo, P.M.: A sem-intrusive determistic apporach to uncertainty quantification in non-linear fluid flow probels. J. Comput. Phys. **235**, 828–845 (2013)
3. Barth, T.: On the propagation of the statistical model parameter uncertainty in CFD calculations. Theor. Comput. Fluid Dyn. (2011). doi:10.1007/s00162-011-0221-2
4. Cockburn, B., Shu, C.W.: The Runge-Kutta discontinuous Galerkin method for conservation laws V: multidimensional systems. J. Comput. Phys. **141**, 199–224 (1998)
5. Debusschere, B., Najm, H., Pébay, P., Knio, O., Ghanem, R., Le Maî tre, O.: Numerical challenges in the use of polynomial chaos representations for stochastic processes. SIAM J. Sci. Comput. **26**, 698–719 (2004)
6. Ghanem, R., Spanos, P.: Stochastic Finite Elements: A Spectral Approach. Dover, Minneola (2003)
7. Godlewski, E., Raviart, P.: Hyperbolic Systems of Conservation Laws. Ellipses Publishing, Paris (1995)
8. Gottlieb, D., Xiu, D.: Galerkin method for wave equations with uncertain coefficients. Commun. Comput. Phys. **3**, 505–518 (2008)
9. Knio, O., Le Maî tre, O.: Uncertainty propagation in CFD using polynomial chaos decomposition. Fluid Dyn. Res. **38**, 616–640 (2006)
10. Le Maî tre, O., Knio, O., Najm, H., Ghanem, R.: Uncertainty propagation using Wiener-Haar expansions. J. Comput. Phys. **197**, 28–57 (2004)
11. Le Maî tre, O., Najm, H., Ghanem, R., Knio, O.: Multi-resolution analysis of Wiener-type uncertainty propagation schemes. J. Comput. Phys. **197**, 502–531 (2004)
12. Le Maî tre, O., Najm, H., Pébay, P., Ghanem, R., Knio, O.: Multi-resolution analysis scheme for uncertainty quantification in chemical systems. SIAM J. Sci. Comput. **29**, 864–889 (2007)
13. LeVeque, R.: Numerical Methods for Conservation Laws. Birkhäuser Verlag, Basel/Boston (1992)
14. Lin, G., Su, C.-H., Karniadakis, G.E.: Predicting shock dynamics in the presence of uncertainties. J. Comput. Phys. **217**, 260–276 (2006)
15. Lin, G., Su, C.-H., Karniadakis, G.E.: Stochastic modelling of random roughness in shock scattering problems: theory and simulations. Comput. Methods Appl. Mech. Eng. **197**, 3420–3434 (2008)
16. Mishra, S., Schwab, Ch.: Sparse tensor multi-level Monte Carlo finite volume methods for hyperbolic conservation laws with random intitial data. Math. Comput. **81**, 1979–2018 (2012)
17. Mishra, S., Schwab, Ch., Šukys, J.: Multi-level Monte Carlo finite volume methods for nonlinear systems of conservation laws in multi-dimensions. J. Comput. Phys. **231**, 3365–3388 (2012)

18. Mishra, S., Risebro, N.H., Schwab, Ch., Tokareva, S.: Numerical solution of scalar conservation laws with random flux functions. SAM report 2012-35. http://www.sam.math.ethz.ch/reports/2012/35
19. Poëtte, G., Després, B., Lucor, D.: Uncertainty quantification for systems of conservation laws. J. Comput. Phys. **228**, 2443–2467 (2009)
20. Schwab, Ch., Tokareva, S.: High order approximation of probabilistic shock profiles in hyperbolic conservation laws with uncertain initial data. ESAIM M2AN **47**, 807–835 (2013)
21. Shu, C.W.: High order ENO and WENO schemes for computational fluid dynamics. In: Barth, T.J., Deconinck, H. (eds.) High-Order Methods for Computational Physics. Springer, Berlin/New York (1999)
22. Troyen, J., Le Maî tre, O., Ndjinga, M., Ern, A.: Intrusive Galerkin methods with upwinding for uncertain nonlinear hyperbolic systems. J. Comput. Phys. **229**, 6485–6511 (2010)
23. Troyen, J., Le Maî tre, O., Ndjinga, M., Ern, A.: Roe solver with entropy corrector for uncertain hyperbolic systems. J. Comput. Phys. **235**, 491–506 (2010)
24. Wan, X., Karniadakis, G.E.: Multi-element generalized polynomial chaos for arbitrary probability measures. SIAM J. Sci. Comput. **28**, 901–928 (2006)
25. Xiu, D., Karniadakis, G.E.: Modeling uncertainty in steady state diffusion problems via generalized polynomial chaos. Comput. Methods Appl. Mech. Eng. **191**, 4927–4948 (2002)
26. Xiu, D., Karniadakis, G.E.: Modeling uncertainty in flow simulations via generalized polynomial chaos. J. Comput. Phys. **187**, 137–167 (2003)

Chapter 8
High-Order Discontinuous Galerkin Solution of Unsteady Flows by Using an Advanced Implicit Method

Alessandra Nigro, Carmine De Bartolo, Francesco Bassi, and Antonio Ghidoni

Abstract The aim of this paper is to investigate and evaluate a multi-stage and multi-step method that is an evolution of the more common Backward Differentiation Formulae (BDF). This new class of formulae, called Two Implicit Advanced Step-point (TIAS), has been applied to a high-order Discontinuous Galerkin (DG) discretization of the Navier-Stokes equations, coupling the high temporal accuracy gained by the TIAS scheme with the high space accuracy of the DG method. The performance of the DG-TIAS scheme has been evaluated by means of two test cases: an inviscid isentropic convecting vortex and a laminar vortex shedding behind a circular cylinder. The advantages of the high-order time discretization are illustrated comparing the sixth-order accurate TIAS scheme with the second-order accurate BDF scheme using the same spatial discretization.

A. Nigro (✉) • C. De Bartolo
Department of Mechanical, Energetic and Management Engineering, University of Calabria, Ponte P. Bucci cubo 44/C, 87036 Rende (CS), Italy
e-mail: alessandra.nigro@unical.it; c.debartolo@unical.it

F. Bassi
Department of Industrial Engineering, University of Bergamo, Viale Marconi 5, 24044 Dalmine (BG), Italy
e-mail: francesco.bassi@unibg.it

A. Ghidoni
Department of Mechanical and Industrial Engineering, University of Brescia, Via Branze 38, 25123 Brescia (BS), Italy
e-mail: antonio.ghidoni@unibs.it

R. Abgrall et al. (eds.), *High Order Nonlinear Numerical Schemes for Evolutionary PDEs*, Lecture Notes in Computational Science and Engineering 99,
DOI 10.1007/978-3-319-05455-1_8,

8.1 Introduction

In recent years the increasing attention to high-order spatial discretization schemes and the continuous growth of computer power put forward the development of high-order temporal methods to perform very accurate and efficient simulations of unsteady flows. In particular, in the context of high-order methods, high accurate time integration schemes are mandatory to capture the significant flow features of transient problems and to perform accurate long time simulations in many areas of research including aeroacoustics, Large Eddy Simulations (LES) and Direct Numerical Simulations (DNS) of turbulent flows.

Some approaches of coupled space-time formulations can be found in the literature [1, 2], but the prevailing number of numerical schemes for the solution of the Euler and the Navier Stokes equations applies the method of lines, employing for the time integration of the space discretized equations one of the methods available for the integration of ordinary differential equations. Among them, explicit Runge-Kutta schemes are easy to implement and parallelize, and require only limited memory storage. However, for problems requiring high spatial resolutions of very thin boundary layers and characterized by very stiff system of equations, the time step restriction would results in an inefficient time integration technique and implicit methods must be used. Among the implicit methods, very popular approaches for the simulation of unsteady flows are the multi-stage implicit Runge-Kutta (RK) schemes [3–5] and the Backward Differentiation Formula (BDF) [6–8]

In implicit multi-stage RK methods the accuracy order can be arbitrarily raised while retaining L-stability by increasing the number of stages, but some family schemes are susceptible to order reduction in the presence of substantial stiffness such as in turbulent flow computations. The implicit multi-step BDF methods are A-stable up to the second-order but only A(α)-stable for order three and higher. Since Gear's book in 1971, different variants of the BDF have proliferated in an attempt to derive a class of multi-step methods characterized by better stability properties and higher-order accuracy. The stability and the accuracy of the BDF methods have been extended by Cash combining in the Extended BDF (EBDF) and successively in the Modified Extended BDF (MEBDF) the multi-step and multi-stage ideas [9, 10]. In particular, the EBDF and MEBDF schemes are A-stable up to order 4 and A(α)-stable up to order 9. Following a similar approach, a new scheme has been proposed based on the use of two super future points guaranteeing an A-stable formula up to order 6 [11, 12]. This new scheme, called Two Implicit Advanced Step-point (TIAS), involves four stages: the first three are predictor stages that use a standard k-step BDF scheme, the last one is a corrector stage that uses an advanced implicit k-step formula of order $k + 1$. The TIAS scheme was presented in [13, 14] and the stability properties of this approach were investigated in detail in [11].

In this work we present numerical results obtained using this advanced multi-step method applied to a high-order DG discretization of the Navier-Stokes equations [15]. The performance of the DG-TIAS scheme has been evaluated by means of two test cases: an inviscid isentropic convecting vortex and a laminar vortex shedding behind a circular cylinder. For both test cases, the accuracy and

the order of convergence of the TIAS scheme has been assessed by performing a temporal refinement study. Furthermore, to clearly illustrate the advantages of the high-order time discretization, the performance of the sixth-order accurate TIAS scheme has been compared with that of the standard second-order accurate BDF scheme using the same spatial discretization.

In the following of the paper the governing equations are presented in Sect. 8.2. Sections 8.3 and 8.4 are devoted to space (DG) and time (TIAS) discretizations, respectively. Numerical results are discussed in Sect. 8.5. Conclusions are reported in Sect. 8.6.

8.2 Governing Equations

The compressible Navier-Stokes equations in conservative form based on the set of conservative variables $\mathbf{q} = [\rho, \rho u, \rho v, \rho E]^T$ are:

$$\frac{\partial \mathbf{q}}{\partial t} + \nabla \cdot \mathbf{F}_c(\mathbf{q}) = \nabla \cdot \mathbf{F}_v(\mathbf{q}, \nabla \mathbf{q}) \ , \tag{8.1}$$

where $\mathbf{F}_c(\mathbf{f}_c, \mathbf{g}_c)$ and $\mathbf{F}_v(\mathbf{f}_v, \mathbf{g}_v)$ are the inviscid and viscous flux vectors respectively, given by:

$$\mathbf{f}_c = \begin{pmatrix} \rho u \\ \rho u^2 + p \\ \rho uv \\ \rho Hu \end{pmatrix}, \mathbf{g}_c = \begin{pmatrix} \rho v \\ \rho vu \\ \rho v^2 + p \\ \rho Hv \end{pmatrix},$$

$$\mathbf{f}_v = \begin{pmatrix} 0 \\ \tau_{xx} \\ \tau_{yx} \\ \tau_{xx}u + \tau_{yx}v - q_x \end{pmatrix}, \mathbf{g}_v = \begin{pmatrix} 0 \\ \tau_{xy} \\ \tau_{yy} \\ \tau_{xy}u + \tau_{yy}v - q_y \end{pmatrix}.$$

In these equations ρ is the fluid density, u and v are the x and y velocity components respectively and p is the pressure. E is the total internal energy for unit mass and the total enthalpy for unit mass is given by $H = E + p/\rho$.

The shear stress tensor components τ_{ij} and the heat flux vector components q_i of viscous flux vectors can be calculated as:

$$\tau_{xx} = \left(2\mu \frac{\partial u}{\partial x} + \lambda \nabla \cdot \mathbf{v}\right), \tau_{yy} = \left(2\mu \frac{\partial v}{\partial y} + \lambda \nabla \cdot \mathbf{v}\right),$$

$$\tau_{xy} = \tau_{yx} = \mu \left(\frac{\partial u}{\partial y} + \frac{\partial v}{\partial x}\right),$$

$$q_x = -\kappa \frac{\partial T}{\partial x}, q_y = -\kappa \frac{\partial T}{\partial y}.$$

In order to close the system of equations, the Navier-Stokes equations must be augmented by algebraic expressions that relate the internal energy E, the pressure p, the dynamic viscosity μ, the second viscosity coefficient λ and the conductivity coefficient κ to the thermodynamic state of the fluid. For an ideal gas, assuming that the fluid satisfies the equation of state of perfect gas, the pressure is given by $p = \rho(\gamma - 1)\left[E - \left(u^2 + v^2\right)/2\right]$, where γ is the ratio of specific heats of the fluid, given by $\gamma = C_p/C_v$. The dynamic viscosity coefficient μ can be approximated using the power-law viscosity formula:

$$\frac{\mu}{\mu_0} = \left(\frac{T}{T_0}\right)^{3/4} .$$

8.3 DG Space Discretization

The BR2 scheme for the DG discretization of (8.1) is presented in [16, 17] and theoretically analyzed in [18,19]. In order to construct the BR2 scheme, we consider an approximation Ω_h of the domain Ω consisting of a set of non-overlapping elements $\tau_h = \{K\}$, denoting by $\partial\Omega_h$ the boundary of the discrete approximation and by Γ_h^0 the set of internal edges. We consider piecewise polynomial functions on τ_h with no global continuity requirement. If $P_n(K)$ denotes the space of polynomial functions of degree at most n in the element K, and considering the function space:

$$\mathbf{V}_h = \{\mathbf{v}_h \in \left(L^2(\Omega_h)\right)^{N+2} : \mathbf{v}_h \in (P_n(K))^{N+2} \; \forall K \in \tau_h\} ,$$

where N is the number of spatial dimensions, the DG formulation of (8.1) is then as follows: find $\mathbf{q}_h \in \mathbf{V}_h$ so that

$$\begin{aligned}
&\int_{\Omega_h} \mathbf{v}_h \cdot \frac{\partial \mathbf{q}_h}{\partial t} \mathrm{d}\mathbf{x} - \int_{\Omega_h} \nabla \mathbf{v}_h : (\mathbf{F}_c(\mathbf{q}_h) - \mathbf{F}_v(\mathbf{q}_h, \nabla \mathbf{q}_h + \mathscr{R}([[\mathbf{q}_h]]_0))) \, \mathrm{d}\mathbf{x} + \\
&\int_{\Gamma_h^0} \left(\mathbf{v}_h^- - \mathbf{v}_h^+\right) \cdot \mathbf{H}\left(\mathbf{q}_h^+, \mathbf{q}_h^-, \mathbf{n}^-\right) \mathrm{d}\sigma - \int_{\Gamma_h^0} [[\mathbf{v}_h]] : \{\mathbf{F}_v(\mathbf{q}_h, \nabla \mathbf{q}_h + \eta_e \mathscr{R}_e([[\mathbf{q}_h]]_0))\} \mathrm{d}\sigma + \\
&\int_{\partial\Omega_h} (\mathbf{v}_h \otimes \mathbf{n}) : \left(\mathbf{H}\left(\mathbf{q}_h^+, \mathbf{q}_h^b, \mathbf{n}\right) - \mathbf{F}_v(\mathbf{q}_h, \nabla \mathbf{q}_h + \eta_e \mathscr{R}_e([[\mathbf{q}_h]]_0))_b\right) \mathrm{d}\sigma = 0 ,
\end{aligned} \tag{8.2}$$

holds for an arbitrary test function $\mathbf{v}_h \in \mathbf{V}_h$.

In this equation η_e is called "penalty" parameter and its lower bound is established as the number of neighbours of the generic element K to guarantee the stability of the method. $\mathscr{R}_e([[\mathbf{q}_h]]_0)$ and $\mathscr{R}([[\mathbf{q}_h]]_0)$ are, respectively, the local and global lifting operators accounting in the gradient of the diffusive fluxes for the jumps in $\mathbf{q}_h$ occurring at the element interfaces, defined as:

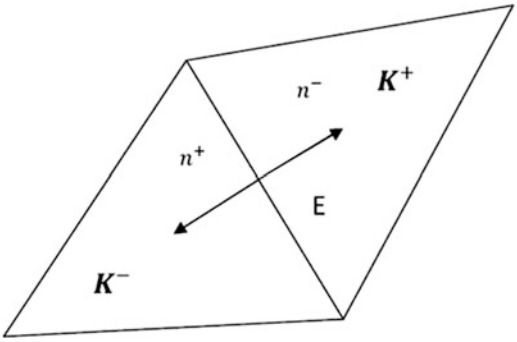


Fig. 8.1 Two elements K^+ and K^- sharing edge E

$$[[\mathbf{q}_h]]_0 = \begin{cases} [[\mathbf{q}_h]] & \text{on } \Gamma_h^0 \\ \mathbf{q}_h \otimes \mathbf{n} & \text{on } \partial\Omega_h \end{cases} ,$$

$$[[\mathbf{q}_h]] = \mathbf{q}_h^+ \otimes \mathbf{n}^+ + \mathbf{q}_h^- \otimes \mathbf{n}^- .$$

Furthermore, the trace operator $\{(\cdot)\}$ denotes the average between left $(\cdot)^-$ and right state $(\cdot)^+$, see Fig. 8.1.

$\mathbf{H}\left(\mathbf{q}_h^+, \mathbf{q}_h^-, \mathbf{n}^-\right)$ and $\mathbf{H}\left(\mathbf{q}_h^+, \mathbf{q}_h^b, \mathbf{n}\right)$ are the numerical flux functions at the interior and boundary faces, respectively. For the inviscid parts of the numerical flux any of the numerical flux functions commonly considered in the finite volume method can be used. In the present work we employ the Godunov flux, i.e. the physical flux of the exact solution of a planar Riemann problem in the direction normal to the interface.

8.4 TIAS Time Discretization

The DG space discretization (8.2) results in the following system of implicit differential equations:

$$\mathsf{M}\frac{\mathrm{d}\mathbf{Q}}{\mathrm{d}t} + \mathbf{R}\left(\mathbf{Q}\right) = 0 , \tag{8.3}$$

where M is the global block diagonal mass matrix, $\mathbf{Q}$ is the global vector of unknown degrees of freedom and $\mathbf{R}\left(\mathbf{Q}\right)$ is the vector of "residuals", i.e., the vector of nonlinear functions of $\mathbf{Q}$ resulting from the integrals of the DG discretized space differential operators in (8.2). Note that M is a constant non-singular matrix.

The Two Implicit Advanced Step-point (TIAS) method is applied to (8.3) to advance the solution in time. Assuming that approximate solutions $\mathbf{Q}_{n+j}$ have been calculated at t_{n+j} with $0 \leq j \leq k-1$, the general k-step TIAS algorithm of order $k+1$ consists of successively solving the following four stages to advance the solution in time:

- Stage 1. Compute the first predictor $\bar{\mathbf{Q}}_{n+k}$ of order k with a k-step BDF:

$$\mathsf{M}\left(\bar{\mathbf{Q}}_{n+k}+\sum_{j=0}^{k-1}\hat{\alpha}_j\mathbf{Q}_{n+j}\right)+\Delta t\hat{\beta}_k\mathbf{R}\left(\bar{\mathbf{Q}}_{n+k}\right)=0\,.$$

- Stage 2. Compute the second predictor $\bar{\mathbf{Q}}_{n+k+1}$ of order k with a k-step BDF:

$$\mathsf{M}\left(\bar{\mathbf{Q}}_{n+k+1}+\hat{\alpha}_{k-1}\bar{\mathbf{Q}}_{n+k}+\sum_{j=0}^{k-2}\hat{\alpha}_j\mathbf{Q}_{n+j+1}\right)+\Delta t\hat{\beta}_k\mathbf{R}\left(\bar{\mathbf{Q}}_{n+k+1}\right)=0\,.$$

- Stage 3. Compute the third predictor $\bar{\mathbf{Q}}_{n+k+2}$ of order k with a k-step BDF:

$$\mathsf{M}\left(\bar{\mathbf{Q}}_{n+k+2}+\hat{\alpha}_{k-1}\bar{\mathbf{Q}}_{n+k+1}+\hat{\alpha}_{k-2}\bar{\mathbf{Q}}_{n+k}+\sum_{j=0}^{k-3}\hat{\alpha}_j\mathbf{Q}_{n+j+2}\right)+$$
$$+\Delta t\hat{\beta}_k\mathbf{R}\left(\bar{\mathbf{Q}}_{n+k+2}\right)=0\,.$$

- Stage 4. Compute the corrected solution $\mathbf{Q}_{n+k}$ of order $k+1$ using:

$$\mathsf{M}\left(\mathbf{Q}_{n+k}+\sum_{j=0}^{k-1}\tilde{\alpha}_j\mathbf{Q}_{n+j}\right)+\Delta t\Big[\tilde{\beta}_{k+2}\mathbf{R}\left(\bar{\mathbf{Q}}_{n+k+2}\right)+\tilde{\beta}_{k+1}\mathbf{R}\left(\bar{\mathbf{Q}}_{n+k+1}\right)+$$
$$+\beta_k\mathbf{R}\left(\bar{\mathbf{Q}}_{n+k}\right)+\left(\tilde{\beta}_k-\beta_k\right)\mathbf{R}\left(\mathbf{Q}_{n+k}\right)\Big]=0\,.$$

In the first three stages $\hat{\alpha}_j$ and $\hat{\beta}_k$ are the BDF coefficients and in the last one $\tilde{\alpha}_j$, $\tilde{\beta}_{k+2}$, $\tilde{\beta}_{k+1}$, $\tilde{\beta}_k$ and β_k are the TIAS coefficients. In particular, $\tilde{\beta}_{k+2}$ and β_k are free coefficients which determine the stability properties of the scheme, while the other coefficients, expressed in terms of $\tilde{\beta}_{k+2}$, are determined such as the scheme has order $k+1$. The residuals in stage 4, depending on the previous stages solutions, are computed as:

$$\mathbf{R}\left(\bar{\mathbf{Q}}_{n+k}\right)=-\frac{\mathsf{M}}{\Delta t\hat{\beta}_k}\left(\bar{\mathbf{Q}}_{n+k}+\sum_{j=0}^{k-1}\hat{\alpha}_j\mathbf{Q}_{n+j}\right)\,,$$

$$\mathbf{R}\left(\bar{\mathbf{Q}}_{n+k+1}\right)=-\frac{\mathsf{M}}{\Delta t\hat{\beta}_k}\left(\bar{\mathbf{Q}}_{n+k+1}+\hat{\alpha}_{k-1}\bar{\mathbf{Q}}_{n+k}+\sum_{j=0}^{k-2}\hat{\alpha}_j\mathbf{Q}_{n+j+1}\right)\,,$$

$$\mathbf{R}\left(\bar{\mathbf{Q}}_{n+k+2}\right)=-\frac{\mathsf{M}}{\Delta t\hat{\beta}_k}\left(\bar{\mathbf{Q}}_{n+k+2}+\hat{\alpha}_{k-1}\bar{\mathbf{Q}}_{n+k+1}+\hat{\alpha}_{k-2}\bar{\mathbf{Q}}_{n+k}+\sum_{j=0}^{k-3}\hat{\alpha}_j\mathbf{Q}_{n+j+2}\right)\,.$$

Table 8.1 Coefficients of the 5-step BDF predictor stages

$\hat{\alpha}_4$	$\hat{\alpha}_3$	$\hat{\alpha}_2$	$\hat{\alpha}_1$	$\hat{\alpha}_0$	$\hat{\beta}_5$
$-300/137$	$300/137$	$-200/137$	$75/137$	$-12/137$	$60/137$

Table 8.2 Coefficients of the 5-step TIAS corrector stage

$\tilde{\alpha}_4$	$\tilde{\alpha}_3$	$\tilde{\alpha}_2$	$\tilde{\alpha}_1$	$\tilde{\alpha}_0$	$\tilde{\beta}_5$	β_5	$\tilde{\beta}_6$	$\tilde{\beta}_7$
$-\frac{26,550}{14,919}+\frac{914,897}{59,676}\cdot\tilde{\beta}_7$	$\frac{18,700}{14,919}-\frac{459,473}{14,919}\cdot\tilde{\beta}_7$	$-\frac{3,200}{4,973}+\frac{11,762}{4,973}\cdot\tilde{\beta}_7$	$\frac{2,925}{14,919}-\frac{123,215}{14,919}\cdot\tilde{\beta}_7$	$-\frac{394}{14,919}+\frac{74,711}{59,676}\cdot\tilde{\beta}_7$	$\frac{2,940}{4,973}+\frac{48,933}{4,973}\cdot\tilde{\beta}_7$	$\frac{1}{10}$	$-\frac{200}{4,973}-\frac{25,961}{4,973}\cdot\tilde{\beta}_7$	$\frac{1}{20}$

Regarding the efficiency of the algorithm, even if four nonlinear systems per time step must be solved, stages 1, 2 and 4 are normally relatively cheap compared with stage 3 because:

- The solution of the second stage at the current time step represents a good approximation to the solution of the first stage at the next time step;
- The solution of the third stage at the current time step represents a good approximation to the solution of the second stage at the next time step;
- The solution of the first stage at each time step represents a good approximation to the solution of the fourth stage.

To solve the nonlinear system of equations at each stage, a Newton iteration scheme is employed where the Jacobian matrix is $\mathbf{J} = \partial\mathbf{R}/\partial\mathbf{Q}$ and the resulting linear algebraic system is solved with the restarted GMRES(m) method. In actual implementation of TIAS schemes we compute the Jacobian matrix only at the first Newton iteration of stage 1 and stage 4 at each time step and recompute it if the convergence of the Newton method becomes too slow. In particular the Jacobian matrix is re-evaluated when the ratio between the L_2 norm of two successive solution variations is less than 5.

Regarding the stability, TIAS method is A-stable up to order 6 and A(α)-stable up to order 9. In this work TIAS method has been investigated to evaluate if the computational effort required to solve the four stages is balanced by its better stability and accuracy compared to other implicit multi-step methods. In particular, the performance of the sixth-order accurate 5-step TIAS scheme has been compared with the performance of the standard second-order accurate BDF scheme. The 5-step BDF coefficients and the 5-step TIAS coefficients are given in Tables 8.1 and 8.2, respectively.

8.5 Numerical Results

This section shows the numerical results of the proposed DG-TIAS scheme for an inviscid (Sect. 8.5.1) and a laminar (Sect. 8.5.2) test case.

The first test case is an inviscid isentropic convecting vortex for which an exact solution is available [20–22]. As the vortex is simply diagonally convected, the exact solution for this test case at any time t is the initial solution at $t_0 = 0$ translated over a distance $\mathbf{v}_\infty t$ on the (x, y) plane. Design-order convergence has been assessed in terms of the L_2-norm of the pressure errors by performing a temporal refinement study.

The second test case is the laminar vortex shedding behind a circular cylinder [23, 24]. The analysis has been performed according to [25–27]. In particular, in order to evaluate the performances of the temporal schemes, a reference solution has been computed with a very small time-step by using the sixth-order accurate TIAS scheme (TIAS6). Design-order convergence has been assessed by performing a temporal refinement study using the lift on the body as an error measure.

For both test cases the accuracy and the efficiency of the TIAS6 scheme is discussed and compared with that of the standard second-order accurate BDF scheme (BDF2) using the same accurate spatial discretization.

In order to properly analyze the performances of the time discretization schemes, the parameters that drive the algebraic solver have to be carefully tuned. The nonlinear system of equations at each stage of BDF2 and TIAS6 schemes has been solved by using the Newton scheme. The resulting linear systems have been iteratively solved using the restarted generalized minimum residual (GMRES) method with the block Jacobi preconditioning available in the PETSc library [28]. A number of preliminary computations on both test cases has shown that a linear-solver normalized-residual tolerance of 10^{-2} allows for efficient computations while the Newton problem should be converged down to one order of magnitude less than the solution error.

All the computations have been performed in parallel using 12 cores on the PLX cluster at CINECA (Intel(R) Xeon(R) Westmere six-core E5645 processor, with a clock of 2.40 GHz).

8.5.1 Convection of an Isentropic Vortex

The uniform stream values of the vortex problem are the flow density ρ_∞, velocity, u_∞ and v_∞, pressure p_∞ and temperature T_∞, which are set as $(\rho_\infty, u_\infty, v_\infty, p_\infty, T_\infty) = (1, \sqrt{\gamma}, \sqrt{\gamma}, 1, 1)$, where $\gamma = 1.4$ is the ratio of specific heats. At $t_0 = 0$ the free stream flow is perturbed by an isentropic vortex $(\delta u, \delta v, \delta T)$ centered at (x_0, y_0) with:

$$\delta u = -\frac{\alpha}{2\pi}(y - y_0)\, \mathrm{e}^{\phi(1-r^2)}\,, \tag{8.4}$$

$$\delta v = \frac{\alpha}{2\pi}(x - x_0)\, \mathrm{e}^{\phi(1-r^2)}\,, \tag{8.5}$$

$$\delta T = -\frac{\alpha^2(\gamma - 1)}{16\phi\gamma\pi^2}\mathrm{e}^{2\phi(1-r^2)}\,, \tag{8.6}$$

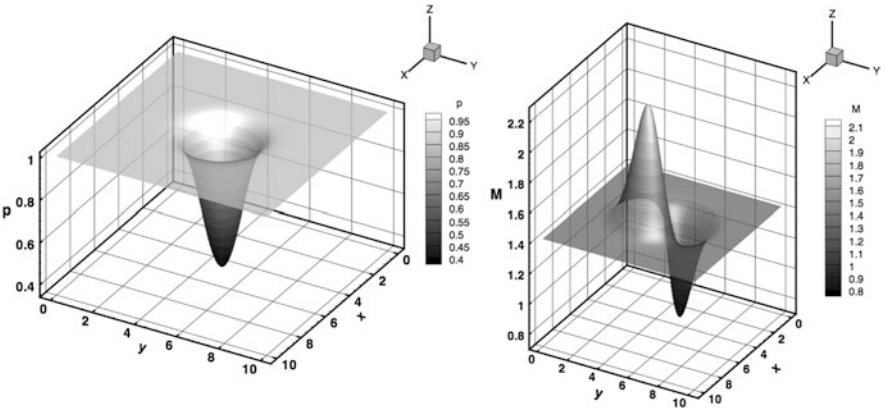


Fig. 8.2 Initial solution of the isentropic vortex problem: pressure (*left*) and Mach number (*right*)

where $r = \sqrt{(x - x_0)^2 + (y - y_0)^2}$ is the distance to the vortex center and ϕ and α are parameters which determine the strength of the vortex. In this study, we set $\phi = \frac{1}{2}$ and $\alpha = 5$. Given the perturbation functions (8.4–8.6) and assuming isentropic flow conditions throughout the domain, the initial solution can be easily determined.

Figure 8.2 shows the pressure (left) and the Mach number (right) contours at time $t_0 = 0$. The vortex is initially placed at $(x_0, y_0) = (5, 5)$ in the domain $0 \leq x \leq 10$ and $0 \leq y \leq 10$ on a uniform cartesian grid with 50×50 quadrangular elements. Periodic boundary conditions are set at top and bottom boundaries and at left and right boundaries, respectively. The analysis is performed up to a final time corresponding to one period T of vortex revolution with progressively finer time steps and a seventh-order space accuracy (P6 elements) to keep the error due to spatial discretization below the temporal error. The notations P6-BDF2 and P6-TIAS6 will be used in the rest of the paper to denote the couple of space-time schemes used.

Considering the dimension of the computational domain and setting periodic boundary conditions, the problem actually solved is that of a system of infinite vortices that move along the diagonal. The initial solution has to take into account of the effect of the four vortices surrounding the central one. This effect is shown in Fig. 8.3 for the pressure (left) and the horizontal velocity component u (right) where pressure and velocity variations with respect to their free stream values are highlighted. If the desired accuracy level is lower than the maximum order of these fluctuations near the boundaries (10^{-6} for the fluctuation of velocity), neglecting the four vortices surrounding the central one leads to a wrong evaluation of the error.

The unknown additional starting values have been computed by using the analytical solution.

The convergence rates of the L_2-norm errors of the pressure field at different time steps are presented in Table 8.3 for both schemes. The table shows that the P6-TIAS6 scheme converges with a maximum value of 5.93, very close to the

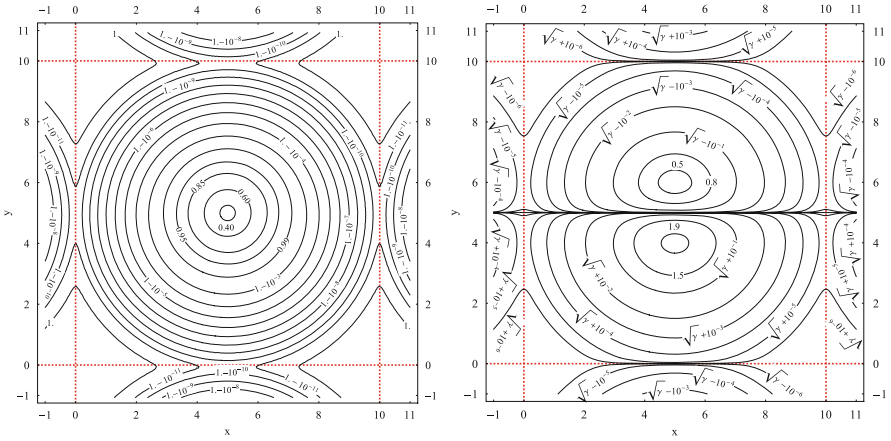


Fig. 8.3 Effect of the imposition of periodic boundary conditions on the initial solution: pressure (*left*) and horizontal velocity component u (*right*)

Table 8.3 L_2-norm errors of the pressure field, orders of convergence and CPU times for different time steps

	P6-TIAS6			P6-BDF2		
$T/\Delta t$	L_2 error(p)	Order	CPU (s)	L_2 error(p)	Order	CPU (s)
20	$1.73 \cdot 10^{-2}$		$6.90 \cdot 10^{3}$	$7.97 \cdot 10^{-2}$		$2.29 \cdot 10^{3}$
		3.52			0.48	
40	$1.51 \cdot 10^{-3}$		$9.94 \cdot 10^{3}$	$5.73 \cdot 10^{-2}$		$3.98 \cdot 10^{3}$
		5.08			1.23	
80	$4.48 \cdot 10^{-5}$		$1.46 \cdot 10^{4}$	$2.44 \cdot 10^{-2}$		$7.31 \cdot 10^{3}$
		5.71			1.85	
160	$8.54 \cdot 10^{-7}$		$2.70 \cdot 10^{4}$	$6.77 \cdot 10^{-3}$		$1.44 \cdot 10^{4}$
		5.93			1.98	
320	$1.40 \cdot 10^{-8}$		$4.33 \cdot 10^{4}$	$1.72 \cdot 10^{-3}$		$1.79 \cdot 10^{4}$
		5.67			2.00	
640	$2.75 \cdot 10^{-10}$		$8.47 \cdot 10^{4}$	$4.31 \cdot 10^{-4}$		$3.30 \cdot 10^{4}$
		–			2.00	
1,280	–		–	$1.08 \cdot 10^{-4}$		$6.57 \cdot 10^{4}$
		–			2.00	
2,560	–		–	$2.69 \cdot 10^{-5}$		$1.32 \cdot 10^{5}$

design accuracy, and that for the lower time step the convergence reduces because the temporal discretization error becomes lower than the spatial discretization one. Conversely, the P6-BDF2 scheme converges with the design accuracy of 2.

In Table 8.3 the accuracy levels achieved by the two schemes are reported as the L_2-norm of the pressure error (L_2 error(p)). The table shows that for a given time step the pressure error of the P6-TIAS6 scheme is significantly smaller than that of the BDF2 scheme and that this difference becomes greater as the time step reduces. This is clearly shown in the left plot of Fig. 8.4 where the L_2 error(p) as a function of the time step is presented. The figure shows that to reach the accuracy level of about $2 \cdot 10^{-2}$ the time step of the P6-TIAS6 is about four times greater than the time step of the P6-BDF2 and that this ratio becomes greater decreasing the error. For example, the order of accuracy of 10^{-5} is achieved by P6-TIAS6 with a time step of $T/80$ whereas P6-BDF2 needs a time step of $T/2{,}560$, 32 times lower than the previous one.

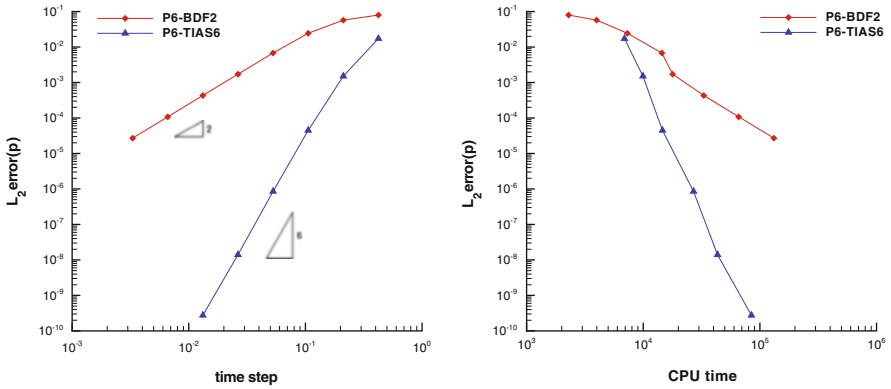


Fig. 8.4 L_2-norm error of the pressure field as a function of the time step (*left*) and of the CPU time (*right*)

In order to compare the efficiency of the temporal schemes in the right plot of Fig. 8.4 the L_2 error(p) of the two schemes as a function of the CPU time are presented. The figure shows that the P6-BDF2 scheme is only competitive with the P6-TIAS6 scheme for high errors (i.e. $>10^{-2}$). As the error decreases, the P6-TIAS6 scheme outperforms the second-order accurate scheme. In particular, the P6-BDF2 scheme requires around 2 times and 8 times the CPU time needed by the P6-TIAS6 scheme to achieve an error of about 10^{-3} and $3 \cdot 10^{-5}$, respectively. For errors lower than 10^{-5} the performance of the P6-TIAS6 scheme, compared with the P6-BDF2 scheme, increases dramatically.

8.5.2 *Unsteady Vortex Shedding Behind a Circular Cylinder*

The second test case is the laminar flow around a two-dimensional circular cylinder computed at Mach number $M_\infty = 0.2$ and Reynolds number $Re = 100$.

The computational grid used in this study, obtained by means of the Gmsh software [29] and shown in Fig. 8.5, is a quadratic mesh containing 3,690 quadrilateral elements with 72 elements lying on the cylinder surface. The wall-distance of the first grid nodes around the cylinder is about 5 % of the cylinder radius. On the wall boundary of the cylinder we impose a zero heat flux no slip boundary condition. For the other boundaries, entropy and stagnation enthalpy are specified at inflow and pressure at outflow.

For this test case an exact solution is not available. Hence a reference solution has been obtained from a TIAS6 run with a very small time step $\Delta t = T/5{,}120$, where T is the vortex shedding period, so that the dominant component of error is the spatial error. In order to evaluate the performance of the temporal schemes, simulations have been performed up to a final time equal to one and half vortex

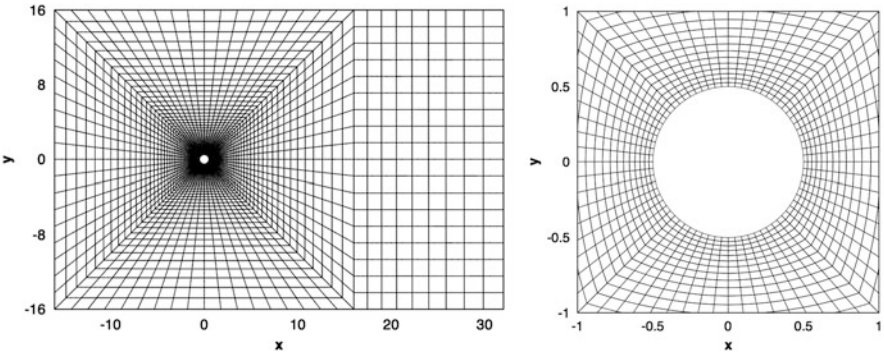


Fig. 8.5 Computational grid for the circular cylinder test case

shedding period for progressively finer time steps. The resulting lift coefficients at the final time have been compared with the reference value computed with the same spatial accuracy (P5 elements). To provide the required solutions to start BDF2 and TIAS6 schemes a general algorithm presented in [30] has been employed. The solution has been computed by using elements ranging from 1st to 6th order. Starting from an initial flow field at rest, a steady P0 solution is computed by means of the backward Euler scheme. Starting from this solution, an unsteady P1 solution is found with $\Delta t = T/10$ by using the TIAS6 scheme. After about 10 regular vortex shedding periods, the solution switches to P2 approximation and continues, using the same integration scheme and the same time step of the P1 solution, for about others 10 vortex shedding periods. Then the P2 solution switches to P3 approximation and so on until a P5 approximation with a time step of $T/160$ was stored in a restart file and used as the initial condition for all the other simulations.

The difference between the computed and the reference lift coefficient, chosen as an error measure, is reported in Table 8.4 together with the convergence rate and CPU time. As shown in this table, the expected design-convergence rates are achieved. To examine the performance of the two temporal schemes, Fig. 8.6 shows the lift error versus time step on the left plot and the lift error versus CPU time on the right plot. The conclusions about the efficiency and the accuracy achieved by the two schemes are similar to those of the inviscid test case. In particular, from Fig. 8.6 it is evident that the P5-BDF2 scheme will never compete with the P5-TIAS6 scheme in terms of efficiency for accuracy levels lower than 10^{-2}. For example, to reach the accuracy level of $\simeq 5 \cdot 10^{-3}$ the P5-BDF2 requires 1.25 times the work needed by P5-TIAS6. Furthermore, for low errors the work required by the P5-BDF2 increases dramatically and in these cases the efficiency of P5-TIAS6 greatly exceeds that of P5-BDF2. For example, the CPU time reduction of P5-TIAS6 with respect to P5-BDF2 is equal to about 30 for an error of $\simeq 10^{-6}$.

Table 8.4 Errors of the lift coefficient, orders of convergence and CPU times for different time steps

	P5-TIAS6			P5-BDF2		
$T/\Delta t$	Error(Lift)	Order	CPU (s)	Error(Lift)	Order	CPU (s)
10	$5.96 \cdot 10^{-3}$		$9.85 \cdot 10^{3}$	$1.43 \cdot 10^{-1}$		$2.92 \cdot 10^{3}$
20	$1.22 \cdot 10^{-3}$	2.28	$1.13 \cdot 10^{4}$	$6.59 \cdot 10^{-2}$	1.12	$5.31 \cdot 10^{3}$
40	$3.83 \cdot 10^{-5}$	5.00	$1.38 \cdot 10^{4}$	$1.99 \cdot 10^{-2}$	1.73	$6.34 \cdot 10^{3}$
80	$5.12 \cdot 10^{-7}$	6.22	$2.78 \cdot 10^{4}$	$5.77 \cdot 10^{-3}$	1.78	$1.21 \cdot 10^{4}$
160	$7.80 \cdot 10^{-9}$	6.04	$6.46 \cdot 10^{4}$	$1.44 \cdot 10^{-3}$	2.00	$2.46 \cdot 10^{4}$
320	–	–	–	$3.57 \cdot 10^{-4}$	2.01	$4.64 \cdot 10^{4}$
640	–	–	–	$8.86 \cdot 10^{-5}$	2.01	$1.01 \cdot 10^{5}$
1,280	–	–	–	$2.21 \cdot 10^{-5}$	2.00	$1.80 \cdot 10^{5}$
2,560	–	–	–	$5.51 \cdot 10^{-6}$	2.00	$3.76 \cdot 10^{5}$
5,120	–	–	–	$1.38 \cdot 10^{-6}$	2.00	$7.07 \cdot 10^{5}$

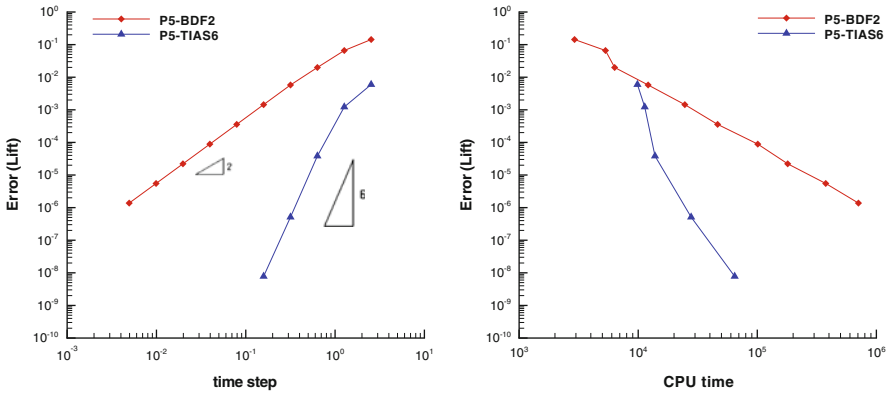


Fig. 8.6 Error of the lift coefficient as a function of the time step (*left*) and of the CPU time (*right*)

8.6 Conclusions

In this paper we have presented a high-order accurate time integration scheme for the numerical solution of the unsteady Navier-Stokes equations. Space discretization is based on the Discontinuous Galerkin (DG) method while time integration employs the A-stable sixth-order accurate Two Implicit Advanced Step-point method (TIAS6). The effectiveness of the scheme has been assessed by computing the convection of an isentropic vortex and the laminar vortex shedding behind a circular cylinder. The computational results show that the TIAS6 scheme provides the design-order of convergence for both test cases. Comparing the computational efficiency of the TIAS6 scheme to that of the popular BDF2 scheme we have found that the latter is more efficient if low accuracy (error $> 10^{-2}$) is required, while for lower error levels TIAS6 outperforms BDF2. The results obtained for smooth-problems highlight the potential of the TIAS method for time

dependent flow computations. However, the cost of the TIAS schemes could be quite large for problems with non-smooth solutions or geometries due to the lack of/worse convergence in the non-linear Newton iterations. Ongoing work is aimed at assessing the effectiveness of TIAS for such kind of applications.

References

1. Klaij, C.M., van der Vegt, J.J.W., Van der Ven, H.: Pseudo-time stepping for space-time discontinuous Galerkin discretizations of the compressible Navier-Stokes equations. J. Comput. Phys. **219**(2), 622–643 (2006)
2. Lörcher, F., Gassner, G., Munz, C.D.: A discontinuous Galerkin scheme based on a space-time expansion. I. Inviscid compressible flow in one space dimension. J. Sci. Comput. **32**(2), 175–199 (2007)
3. Alexander, R.: Diagonally implicit Runge-Kutta methods for stiff O.D.E.s. SIAM J. Numer. Anal. **14**(6), 1006–1021 (1977)
4. Calvo, M.P., de Frutos, J., Novo, J.: Linearly implicit Runge-Kutta methods for advection-reaction-diffusion equations. Appl. Numer. Math. **37**(4), 535–549 (2001)
5. Kennedy, C.A., Carpenter, M.H.: Additive Runge-Kutta schemes for convection-diffusion-reaction equations. Appl. Numer. Math. **44**(1–2), 139–181 (2003)
6. Curtiss, C.F., Hirschfelder, J.O.: Integration of stiff equation. Proc. Natl. Acad. Sci. USA **38**, 235–243 (1952)
7. Ascher, U., Petzold, L.: Computer Method for Ordinary Differential Equations and Differential Algebraic Equations. Society for Industrial and Applied Mathematics (SIAM), Philadelphia (1998)
8. Gear, W.: Simultaneous numerical solution of differential algebraic equation. IEEE Trans. Circuit Theory **18**, 89–95 (1971)
9. Cash, J.R.: On the integration of stiff systems of O.D.E.s using extended backward differentiation formulae. Numer. Math. **34**, 235–246 (1980)
10. Cash, J.R.: The integration of stiff initial value problems in ODEs using modified extended backward differentiation formulae. Comput. Math. Appl. **9**(5), 645–657 (1983)
11. Psihoyios, G.Y.: A general formula for the stability functions of a group of implicit advanced step-point (IAS) methods. Math. Comput. Model. **46**(1–2), 214–224 (2007). (Elsevier, ISSN 08957177)
12. Cash, J.R.: Modified extended backward differentiation formulae for the numerical solution of stiff initial value problems in ODEs and DAEs. J. Comput. Appl. Math. **125**, 117–130 (2000)
13. Psihoyios, G.Y.: Advanced step-point methods for the solution of initial value problems. PhD thesis, Imperial College of Science and Technology, University of London. 29 Dec 1995
14. Psihoyios, G.Y., Cash, J.R.: A stability result for general linear methods with characteristic function having real poles only. BIT Numer. Math. **38**(3), 612–617 (1998). (Springer, ISSN 00063835)
15. Bassi, F., Rebay, S.: GMRES discontinuous Galerkin solution of the compressible Navier-Stokes equations. Lect. Notes Comput. Sci. Eng. **11**, 197–208 (2000). Springer, Berlin
16. Bassi, F., Rebay, S., Mariotti, G., Pedinotti, S., Savini, M.: A high-order accurate discontinuous finite element method for inviscid and viscous turbomachinery flows. In: Decuypere, R., Dibelius, G. (eds.) Proceeding of the 2nd European Conference on Turbomachinery Fluid Dynamics and Thermodynamics, Antwerpen, 5–7 Mar 1997, pp. 99–108. Technologisch Instituut
17. Bassi, F., Rebay, S.: A high order discontinuous Galerkin method for compressible turbulent flows. In: Cockburn, B., Karniadakis, G., Shu, C.-W. (eds.) Discontinuous Galerkin, vol. 11, pp. 77–88. Springer, Berlin/New York (2000)

18. Brezzi, F., Manzini, M., Marini, D., Pietra, P., Russo, A.: Discontinuous Galerkin approximations for elliptic problems. Numer. Methods Partial Differ. Equ. **16**, 365–378 (2000)
19. Arnold, D.N., Brezzi, F., Cockburn, B., Marini, D.: Unified analysis of discontinuous Galerkin methods for elliptic problems. SIAM J. Numer. Anal. **39**(5), 1749–1779 (2002)
20. Hu, C., Shu, C.W.: Weighted essentially non-oscillatory schemes on triangular meshes. J. Comput. Phys. **150**, 97–127 (1999)
21. Yee, H.C., Sandham, N.D., Djomehri, M.J.: Low dissipative high order shock-capturing methods using characteristic-based filters. J. Comput. Phys. **150**, 199–238 (1999)
22. Wang, L., Mavriplis, D.J.: Implicit solution of the unsteady Euler equations for high-order accurate discontinuous Galerkin discretizations. J. Comput. Phys. **225**, 1994–2015 (2007)
23. Liang, C., Premasuthan, S., Jameson, A.: High-order accurate simulation of low-Mach laminar flow past two side-by-side cylinders using spectral difference method. Comput. Struct. **87**, 812–827 (2009)
24. Meneghini, J.R., Saltara, F., Siqueira, C.L.R., Ferrari, J.A.: Numerical simulation of flow interference between two circular cylinders in tandem and side-by-side arrangements. J. Fluids Struct. **15**, 327–50 (2001)
25. Bijl, H., Carpenter, M.H., Vatsa, V.N.: Time integration schemes for the unsteady Navier-Stokes equations. In: 15th Computational Fluid Dynamics Conference, Anaheim, 11–14 June 2001, pp. 1–12. AIAA 2001–2612 (2001)
26. Bijl, H., Carpenter, M.H., Vatsa, V.N., Kennedy, C.A.: Implicit time integration schemes for the unsteady compressible Navier-Stokes equations: Laminar flow. J. Comput. Phys. **179**, 313–329 (2002)
27. Carpenter, M.H., Kennedy, C.A., Bijl, H., Viken, S.A., Vatsa, V.N.: Fourth-order Runge-Kutta schemes for fluid mechanics applications. J. Sci. Comput. **25**(1), 157–194 (2005)
28. Balay, S., Buschelman, K., Gropp, W.D., Kaushik, D., Knepley, M.G., Curfman Mclnneses, L., Smith, B.F., Zhang, H.: PETSc Web page. http://www.mcs.anl.gov/petsc (2001)
29. Geuzaine, C., Remacle, J.F.: Gmsh: a three-dimensional finite element mesh generator with built-in pre- and post processing facilities. Int. J. Numer. Methods Eng. **79**(11), 1309–1331 (2009)
30. Tirani, R., Paracelli, C.: An Algorithm for starting multistep methods. Comput. Math. Appl. **45**, 123–129 (2003)

Chapter 9
Detecting Edges in High Order Methods for Hyperbolic Conservation Laws

Martina Wirz

Abstract In this paper the edge detection technique based on generalized conjugated Fourier partial sums is extended to the fully twodimensional case using concentration kernels. Several examples with different kernels are provided which show the improved edge detection property. Furthermore, one possibility how to apply this approach to a local method, namely the Spectral-Difference method with modal filtering in the Proriol-Koornwinder-Dubiner basis, is presented and compared with a common coefficient based shock indicator. Some examples which show a better shock indication using the new Fourier based approach are provided.

9.1 Introduction

The detection of discontinuities in the numerical solution of a given conservation law is an important tool to handle shocks or discontinuous initial data appropriately. Up to now there exist various approaches to find discontinuities in a numerical solution. One famous example are 3×3 masks which approximate the gradient and are widely used in image processing since there data is usually given at equidistantly distributed pixels. Another approach is to measure jumps across cell boundaries which is commonly used in Finite Volume methods. If we now consider *high order* local methods, the data is typically distributed not equidistantly such that other shock indicators, as for example measuring oscillations in the leading coefficients of the reconstructed solution, are more feasible. We will focus on yet another edge detector based on the conjugated Fourier partial sum which has shown excellent behaviour in the context of spectral methods. This detector is based on

M. Wirz (✉)
Institut Computational Mathematics, Technische Universität Braunschweig, Pockelsstr. 14, 38106 Braunschweig, Germany
e-mail: m.wirz@tu-bs.de

R. Abgrall et al. (eds.), *High Order Nonlinear Numerical Schemes for Evolutionary PDEs*, Lecture Notes in Computational Science and Engineering 99,
DOI 10.1007/978-3-319-05455-1_9,

the property of the onedimensional conjugated Fourier partial sum of a function f to converge pointwise to the jump height of f, which is often referred to as concentration property. To accelerate the convergence rate one can exploit the fact that the conjugated Fourier partial sums can be written as a convolution

$$\tilde{S}_n f(x) = f * \tilde{D}_n(x) = \int f(t)\tilde{D}_n(x-t)\,\mathrm{d}t \tag{9.1}$$

with the conjugated Dirichlet kernel $\tilde{D}_n$. Considering generalized kernels $\tilde{K}_n$ with similar properties as the Dirichlet kernel proposed in [4], it can be proven that for such generalized conjugated partial sums

$$\tilde{S}^{K_n} f(x) := f * \tilde{K}_n(x) \tag{9.2}$$

the concentration property holds, either. The convergence rate for this generalized sums is improved away from the discontinuity, hence an efficient edge detection with less Fourier coefficients is possible. This has been validated in several testcases in the context of spectral methods in one dimension and extended to the quasi-twodimensional approach (with one variable fixed) in [5].

We will show that this edge detection technique can be extended to the fully twodimensional case considering the (generalized) conjugated Fourier partial sums in two variables, where the partial sums now converge to the jumps in the mixed partial derivatives (compare [10] for the classical conjugated partial sums). Furthermore, we will discuss one procedure how to apply this Fourier based detector to local methods which, to the knowledge of the author, has not been done so far. To this end, we shortly review the Spectral Difference method on triangles with modal filtering [9] w.r.t. the Proriol-Koornwinder-Dubiner basis. Here, the coefficients are modified as in the Spectral Viscosity approach in order to stabilize the method in the presence of shocks or step gradients. For an efficient implementation, we present a formulation to compute the Fourier coefficients directly from the given modal coefficients. We finally apply this edge detection approach to the proposed Spectral Difference method and validate in several testcases that the Fourier based detector yields superior results compared to a common coefficient based shock indicator.

9.2 Conjugated Fourier Partial Sums

Consider the conjugated Fourier partial sum

$$\tilde{S}_n f(x) = -\mathrm{i} \sum_{|k|\leq n} \operatorname{sgn}(k)\hat{\mathbf{f}}_k \mathrm{e}^{\mathrm{i}kx} \tag{9.3}$$

of the function f where $\hat{\mathbf{f}}_k = \dfrac{1}{2\pi}\displaystyle\int_{-\pi}^{\pi} f(t)\mathrm{e}^{-\mathrm{i}kt}\,\mathrm{d}t$ are the Fourier coefficients of f. In this case the following Theorem proven by Lukács [8] in 1920 holds which we refer to as the *concentration property*.

Theorem 1. *Let $f : [-\pi,\pi) \subseteq \mathbf{R} \to \mathbf{R}$ Lebesgue integrable and 2π-periodic. If $\lim\limits_{\varepsilon\to 0} f(x_0-\varepsilon) - f(x_0+\varepsilon) =: d_{x_0}(f)$ exists in $x_0 \in [-\pi,\pi)$, then*

$$\lim_{n\to\infty} -\frac{\pi}{\ln(n)}\tilde{S}_n f(x) = d_{x_0}(f). \tag{9.4}$$

Note that for continuous f in x_0 it holds $d_{x_0}(f) = 0$, hence the conjugated Fourier partial sum can particularly serve as an edge detector. The disadvantage here is a relatively slow convergence rate of $\mathscr{O}\left(\frac{1}{\ln(n)}\right)$, such that many Fourier coefficients would have to be computed to recover the underlying discontinuities.

The conjugated Fourier partial sum can also be written as the convolution

$$\tilde{S}_n f(x) = f * \frac{1}{\pi}\tilde{D}_n(x) = \frac{1}{\pi}\int_{-\pi}^{\pi} f(t)\tilde{D}_n(t-x)\,\mathrm{d}t \tag{9.5}$$

with the conjugated Dirichlet kernel

$$\tilde{D}_n(t) = \frac{\cos\left(\frac{t}{2}\right) - \cos\left(\left(n+\frac{1}{2}\right)t\right)}{2\sin\left(\frac{t}{2}\right)}. \tag{9.6}$$

To accelerate the convergence rate, Gelb and Tadmor [4] extended Lukács result using so-called admissible concentration kernels $\tilde{K}_n$ with similar properties as $-\dfrac{1}{\ln(n)}\tilde{D}_n$.

Definition 1. A conjugated kernel $\tilde{K}_n : [-\pi,\pi] \to \mathbf{R}$ is called admissible, if

- $\tilde{K}_n$ is odd,
- $\lim\limits_{n\to\infty}\displaystyle\int_0^{\pi} \tilde{K}_n(x)\,\mathrm{d}x = -1$,
- $\tilde{K}_n(x) = c\cdot\dfrac{\cos\left((n+\frac{1}{2})x\right)}{2\pi\sin\left(\frac{x}{2}\right)} + \tilde{R}_n(x)$ with $\left\|\tilde{R}_n(x)\right\|_{L^1} \le 1$,
- For all $\delta > 0$ it holds $\lim\limits_{n\to\infty}\sup\limits_{|x|>\delta>0}\left|\tilde{R}_n(x)\right| = 0$.

Theorem 2. *Let f be piecewise smooth, $J = \{\xi \mid d_\xi(f) \neq 0\}$ and $\tilde{K}_n$ an admissible kernel, then it holds*

$$\tilde{S}_n^K f(x) = f * \tilde{K}_n(x) = \int_{-\pi}^{\pi} f(t)\tilde{K}_n(t-x)\,\mathrm{d}t \to d_x(f)\delta_J(x). \tag{9.7}$$

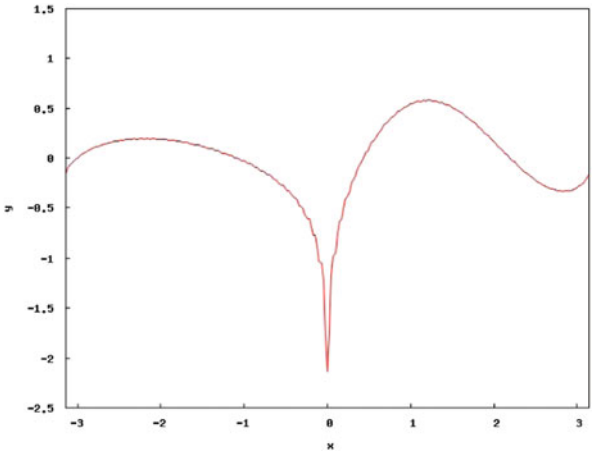


Fig. 9.1 $\tilde{S}_n^D f(x)$ without concentration kernel

Depending on the chosen concentration kernel, improved convergence rates away from discontinuities are possible, e.g. $\mathcal{O}(n^{-p})$ for polynomial kernels as has been shown in [4]. The following example illustrates the advantage of using concentrated Fourier partial sums.

Example 1. Lets consider the periodic function $f : [-\pi, \pi] \to \mathbf{R}$ defined by

$$f(x) = \begin{cases} \sin \frac{x+\pi}{2}, & -\pi \le x < 0, \\ \sin \frac{3x-\pi}{2}, & 0 \le x \le \pi. \end{cases} \tag{9.8}$$

This function is piecewise smooth with a jump of the height -2 in $x_0 = 0$. Figure 9.1 shows the classical conjugated Fourier partial sum with $n = 80$. One can observe that the conjugated partial sum approximates the desired jump function which in this example is -2 at $x = 0$ and zero elsewhere, but at a very poor rate. The concentrated Fourier partial sum with a polynomial concentration kernel shows a much better approximation using the same amount of Fourier coefficients, as can be seen in Fig. 9.2.

9.3 The Generalized Twodimensional Case

The conjugated Fourier partial sum in two dimensions is not unique anymore such that we can consider the conjugation w.r.t. one variable

$$\tilde{S}_{mn}^x f(x, y) = -\mathrm{i} \sum_{|j| \le m} \sum_{|k| \le n} \operatorname{sgn}(j)\, \hat{\mathbf{f}}_{jk} \mathrm{e}^{\mathrm{i}(jx+ky)}, \tag{9.9}$$

respectively $\tilde{S}_{mn}^y f(x, y)$, or w.r.t. two variables

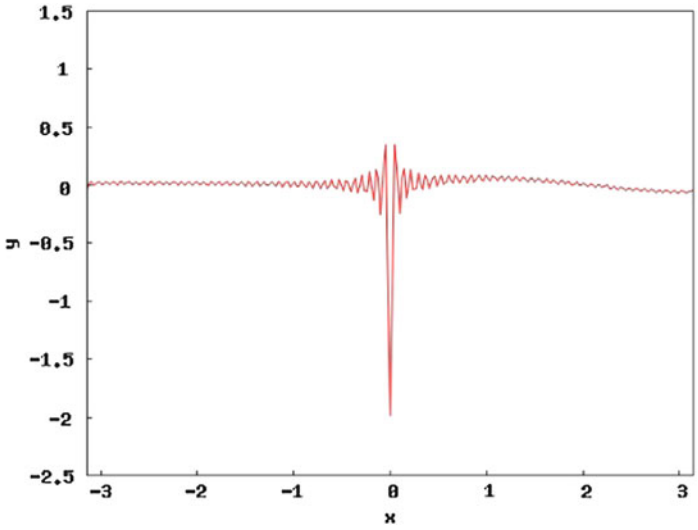


Fig. 9.2 $\tilde{S}_n^K f(x)$ with polynomial concentration kernel

$$\tilde{S}_{mn}(f)(x,y) = -\sum_{|\xi|\le m}\sum_{|\eta|\le n} \operatorname{sgn}(\xi)\operatorname{sgn}(\eta)\,\hat{\mathbf{f}}_{\xi\eta} e^{\mathrm{i}(\xi x+\eta y)}. \tag{9.10}$$

In case of the conjugation w.r.t. one variable, the generalized concentration property in Theorem 2 is directly transferable. In case of two variables, the following theorem holds, see Móricz [10].

Theorem 3. *If* $f \in L^1([-\pi,\pi)^2)$ *and*

$$\lim_{u,v\to 0^+} [f(x-u,y-v)+f(x+u,y+v)-(f(x+u,y-v)+f(x-u,y+v))] =: d_{xy}(f)$$

exists in $(x,y)\in[-\pi,\pi)^2$, *then*

$$\lim_{m,n\to\infty} \frac{\pi^2}{\ln(m)\ln(n)} \tilde{S}_{mn} f(x,y) = d_{xy}(f). \tag{9.11}$$

Here, one does not detect the jumps of the function f in x- resp. y-direction, but the jumps in the mixed partial derivatives, which follows from the definition of $d_{xy}(f)$. Theorem 3 can also be extended to the generalized case using appropriate concentration kernels in two dimensions, see also [14].

Theorem 4. *If* f *is a piecewise smooth function,*

$$J := \{(x,y)\in\mathbf{R}^2 \mid f \text{ discontinuous in } (x,y)\} \tag{9.12}$$

and $\tilde{K}_{mn}(x,y) = \tilde{K}_m(x)\tilde{K}_n(y)$ *with admissible kernels* $\tilde{K}_m$, $\tilde{K}_n$, *then it holds*

$$\tilde{S}^K_{mn} f(x,y) := f * \tilde{K}_{mn}(x,y) = \int_{[-\pi,\pi)^2} f(u,v)\tilde{K}_{mn}(x-u, y-v)\,\mathrm{d}u\,\mathrm{d}v \to d_{xy}(f)\delta_J(x,y). \tag{9.13}$$

Since the usage of this kernels is not very practical in applications, an alternative approach where the summands of the conjugated partial sum w.r.t. two variables are multiplied with concentration factors σ_{jm}, σ_{kn} is useful, i.e.

$$\tilde{S}^\sigma_{nm} f(x) = -\sum_{|j|\le m}\sum_{|k|\le n} \sigma_{jm}\sigma_{kn}\mathrm{sgn}(k)\mathrm{sgn}(j)\hat{\mathbf{f}}_{jk}\mathrm{e}^{\mathrm{i}(jx+ky)}. \tag{9.14}$$

It can be proven that each kernel K_{sigma} is connected to a concentration factor σ via

$$\tilde{K}^\sigma_n(x) = \frac{1}{\pi}\sum_{k=1}^{n} \sigma\left(\frac{k}{n}\right)\sin(kx), \tag{9.15}$$

i.e. $\tilde{S}^{K_\sigma}_{mn} f(x) = \tilde{S}^\sigma_{nm} f(x)$. Hence, the concentration property holds either in this case. The advantage of this representation is that it can easily be extended to the discrete case with modified factors which avoid additional oscillations if only discrete Fourier coefficients are given, which has been realized in one dimension in [5].

This approach can also be extended to the twodimensional case: consider the *discrete* conjugated Fourier partial sum in two variables

$$\tilde{T}^\tau_{mn} f(x,y) = -\sum_{j=-m}^{m}\sum_{k=-n}^{n} \mathrm{sgn}(j)\,\mathrm{sgn}(k)\,\tau_{jk,mn}\tilde{f}_{jk}\mathrm{e}^{\mathrm{i}(jx+ky)} \tag{9.16}$$

with discrete Fourier coefficients

$$\tilde{f}_{jk} = \frac{1}{(2m+1)(2n+1)}\sum_{\mu=-m}^{m}\sum_{\nu=-n}^{n} f(x_\mu, y_\nu)\mathrm{e}^{-\mathrm{i}(jx_\mu+ky_\nu)}. \tag{9.17}$$

Then the following theorem holds [14].

Theorem 5. *Let* $f : \mathbf{R}^2 \to \mathbf{R}$ *be piecewise smooth in each of its two components, and* J *the set of its jump discontinuities. Consider admissible kernels* $\sigma_{j,m}$, $\sigma_{k,n}$ *and define the discrete concentration factor* $\tau_{jk,mn} = \tau_{j,m}\tau_{k,n}$ *given as*

$$\tau_{j,m} = \frac{\sin\left(j\frac{\Delta x}{2}\right)}{j\frac{\Delta x}{2}}\sigma_{j,m} \qquad \text{where } \Delta x = \tfrac{2\pi}{2m+1}, \tag{9.18}$$

$$\tau_{k,n} = \frac{\sin\left(k\frac{\Delta y}{2}\right)}{k\frac{\Delta y}{2}}\sigma_{k,n} \qquad \text{where } \Delta y = \tfrac{2\pi}{2n+1}. \tag{9.19}$$

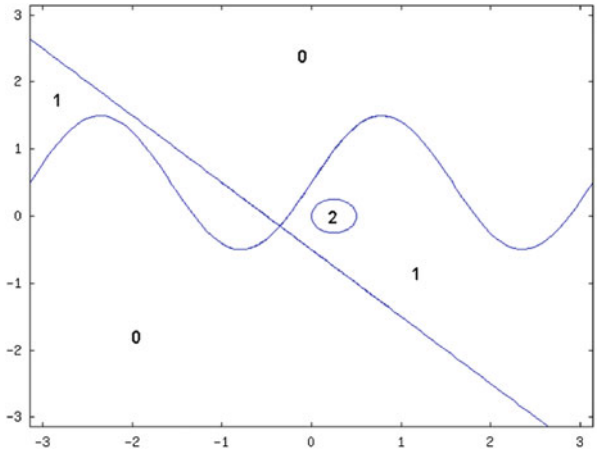


Fig. 9.3 Testcase Example 2: a piecewise constant function f with jumps along $y = -0.5 - x$, $y = 0.5 + \sin(2x)$ and $(x - 0.25)^2 + y^2 = 0.25^2$

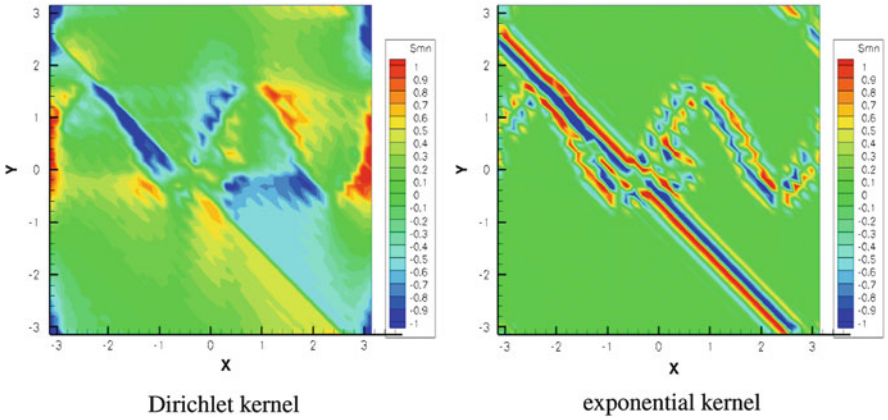


Fig. 9.4 $\tilde{\mathcal{F}}_{nn} f$ for various kernels and discrete Fourier coefficients, $n = 20$, for the piecewise constant function f given in Example 2

Then, $\tilde{T}^{\tau}_{mn} f$ fulfills the concentration property

$$\tilde{T}^{\tau}_{mn} f(x, y) \overset{m,n\to\infty}{\longrightarrow} d_{xy}(f)\delta_J(x, y). \tag{9.20}$$

We will consider the following example with different kernels where the acceleration of the convergence can be seen in two dimensions.

Example 2. Lets consider a piecewise constant function defined as in Fig. 9.3. This function has jumps along $y = -0.5 - x$, $y = 0.5 + \sin(2x)$ and $(x - 0.25)^2 + y^2 = 0.25^2$. In Fig. 9.4, the twodimensional conjugated Fourier partial sum in two

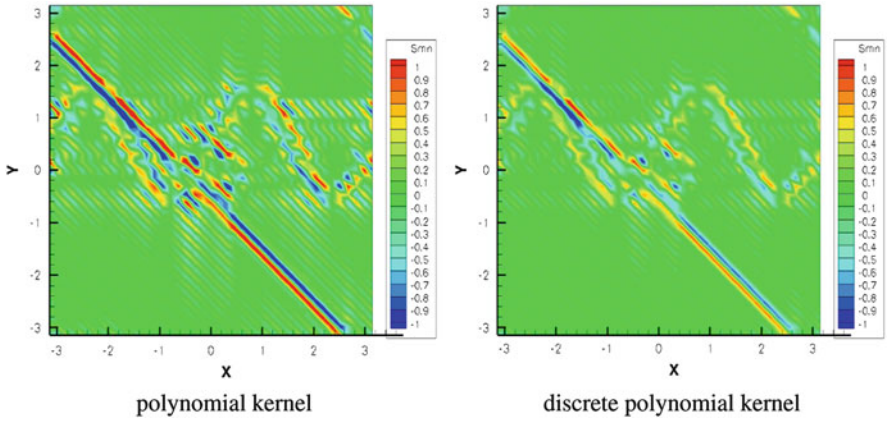


Fig. 9.5 $\tilde{\mathscr{S}}_{nn} f$ for various kernels and discrete Fourier coefficients, $n = 20$, for the piecewise constant function f given in Example 2

variables with $n = 20$ is shown either for the classical Dirichlet kernel and for an exponential kernel proposed in [5]. One can observe the improved detection in case of the exponential kernel since the jumps are much sharper detected. Nevertheless, regions where the mixed partial derivatives are zero cannot be detected (which is clear from the definition of d_{xy}), especially jumps across the bisectors of the x- and y-axis.

Figure 9.5 shows the conjugated Fourier partial sums in two variables with $n = 20$ for the polynomial kernel given by the concentration factor $\sigma(x) = -\pi x$ and its discrete equivalent. Here, the reduced oscillations using the discrete approach in case of discrete Fourier coefficients are visible.

9.4 Application to Local Methods

In order to use the proposed edge detection in local methods, that means in numerical methods for PDEs which reconstruct the unknown variable(s) in local cells and not directly in the whole domain, one meets the following challenges. The data is typically given in modal coefficients w.r.t. a given basis (e.g. modal Discontinuous Galerkin (DG) methods) or at discrete points which are usually interpolation or integration points of the given cell (e.g. nodal DG or Spectral Difference (SD) method). Contrary, discrete Fourier coefficients have to be computed on equidistantly distributed points or using a non-equidistant Fourier approach. To avoid the problematic interpolation, a direct computation of the Fourier coefficients from given coefficients resp. nodal values would be useful. We will first shortly review the Spectral Difference method with modal filtering based on Proriol-

Koornwinder-Dubiner (PKD) polynomials and afterwards present a possibility how the new edge detector can be applied in this context.

9.4.1 The Spectral Difference Method with Modal Filtering

The basic idea of this method (which has been proposed in [7, 13]) is to discretize the right-hand side of the underlying conservation law

$$u_t(\mathbf{x}, t) = -\nabla \cdot \mathscr{F}(u(\mathbf{x}, t)) \tag{9.21}$$

at certain solution points $\mathbf{x}_j$ on each triangular element. Then, one can use an arbitrary explicit time-stepping scheme (like Runge-Kutta) to solve the resulting ODE (in t) at each $\mathbf{x}_j$. To obtain a method of order n, that means to have an exact formulation for $u \in \mathbf{P}_{n-1}$, we need to be exact for $\mathscr{F} \in \mathbf{P}_n^2$ since the derivatives of $\mathscr{F}$ are used to update the solution u.

To approximate the flux $\mathscr{F}$ we reconstruct it in each cell V_i of our triangulation $\mathscr{T}$ w.r.t. an orthogonal basis $\{\phi_k \mid k \in I_{N_F}\}$ on a reference triangle such that

$$\mathscr{F}(u(\mathbf{x}, t) = \begin{pmatrix} F_1(\mathbf{x}, t) \\ F_2(\mathbf{x}, t) \end{pmatrix} = \sum_{k \in N_F} \begin{pmatrix} \hat{F}_{k,1}(t) \\ \hat{F}_{k,2}(t) \end{pmatrix} \phi_k(\mathbf{x}), \tag{9.22}$$

where $F_i(\mathbf{x})$ is the i-th component of the fluxvector. In the original approach Lagrange polynomials L_k with corresponding coefficients $\hat{F}_{k,i}(t) = F_i(\mathbf{x}_k, t)$ were considered.

Inserting (9.22) into (9.21) yields, due to linearity,

$$u_t(\mathbf{x}, t) = -\sum_{k=1}^{N_F} \begin{pmatrix} \hat{F}_{k,1}(t) \\ \hat{F}_{k,2}(t) \end{pmatrix} \cdot \nabla \phi_k(\mathbf{x}). \tag{9.23}$$

Introducing an coordinate transform Λ which maps an arbitrary triangle to the reference triangle, we can rewrite the gradient of ϕ_k as $\nabla \phi_k = \mathscr{J}\Lambda \nabla_{r,s} \phi_k$, where $\mathscr{J}\Lambda$ is the Jacobian of Λ. Thus, we get the universal update scheme

$$u_t(\mathbf{x}, t) = -\sum_{k=1}^{N_F} \begin{pmatrix} \hat{F}_{k,1}(t) \\ \hat{F}_{k,2}(t) \end{pmatrix} \cdot \mathscr{J}\Lambda \nabla_{r,s} \phi_k(r, s) \tag{9.24}$$

where for each cell only the coefficient matrix $\mathscr{J}\Lambda$ has to be stored.

In spite of the Lagrange reconstruction approach where the flux coefficients are directly known we now have to compute $\hat{F}_{k,i}(t)$, $i = 1, 2$ from the given data. If we want to use other basis polynomials than Lagrange polynomials, the coefficients w.r.t. the basis have to be computed from the given nodal data.

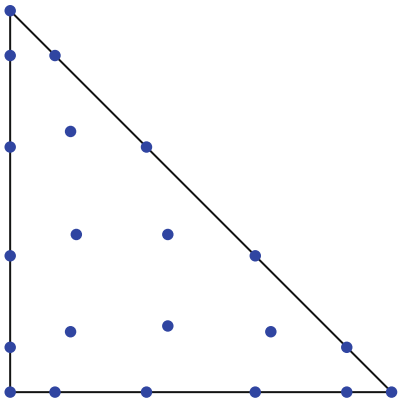


Fig. 9.6 Two dimensional Lobatto grid for SD method, $n = 5$

One possibility is to use projection to achieve $\hat{F}_{k,i}(t) = \frac{1}{\|\phi_k\|^2}\langle F_i, \phi_k \rangle$ in each timestep t. Since then quadrature rules of order $2n$ are needed which require many quadrature points, we choose the interpolation approach instead. Starting from Eq. 9.22, we solve the system of equations $\mathbf{F_i} = \mathbf{V} \cdot \hat{\mathbf{F}}_\mathbf{i}$ where $\mathbf{F} = \left(F_i(\mathbf{x}_j, t)\right)_j$, $\hat{\mathbf{F}}_\mathbf{i} = \left(\hat{F}_{k,i}(t)\right)_k$ for each component $i = 1, 2$, and $\mathbf{V} = \left(\phi_k(\mathbf{x}_j)\right)_{k,j}$ is the Vandermonde matrix. Here, good interpolationpoints have to be chosen which lead to a good conditioning of the matrix $\mathbf{V}$ which has to be inverted. Among the various sets of interpolation points as Fekete, electrostatic points etc., we choose a two dimensional Lobatto points extension (Fig. 9.6) proposed in [2] arising from a geometrical point of view. These points are pretty easy to implement and have a very good Lebesgue constant which indicates good interpolation properties. They are also used as the set of solution points $\mathbf{x}_j$ where the solution is updated such that no additional reconstruction for the flux computation at the flux points is needed.

The coupling between two neighboured cells is provided by replacing the flux values at the flux points $\mathbf{x}$ on the edges by a numerical flux $\mathscr{F}^{num}(\mathbf{x})$ whose normal component $\mathscr{F}^{num}(\mathbf{x}) \cdot \mathbf{n}$ is determined by solving the Riemann problem at the cell interface. For flux points at the vertices, we obtain the numerical flux by solving two Riemann problems in $\mathbf{n}_l$ and $\mathbf{n}_r$ direction, whereas for flux points at the edge a second condition to determine the whole flux vector is free to choose. In this work, we set the tangential component of the numerical flux to remain the same as in the original flux, that means $\mathscr{F}^{num}(\mathbf{x}) \cdot \mathbf{t} = \mathscr{F}(\mathbf{x}) \cdot \mathbf{t}$.

In our numerical examples, we will use modal filtering to stabilize the Spectral Difference method in the presence of shocks or steep gradients. In the context of spectral methods which are known to lack sufficient dissipation, Tadmor [11] proposed the Spectral Viscosity method which solves Eq. (9.21) with an additional viscosity term on the right-hand side in Fourier space. Applied to the underlying conservation law, this results in filtered coefficients $\hat{u}_k^\sigma = \sigma(\theta)\hat{u}_k$ where σ is an exponential filter based on a differential operator arising from the viscosity

term. Since we want to apply this approach to the Spectral Difference method on triangulations, we choose an orthogonal basis on the reference triangle

$$\mathbf{T}^2 := \{(r,s) \in \mathbf{R}^2 : r,s \geq -1, r+s \leq 0\}, \tag{9.25}$$

namely the Proriol-Koornwinder-Dubiner (PKD) polynomials (compare [3, 6]). They result as a kind of 'warped product' from transforming the unit square $[-1,1]^2$ to the reference triangle $\mathbf{T}^2$ (see e.g. [14] for details). Since the Jacobi polynomials $P_n^{\alpha,\beta}$ are known as an orthogonal and stable basis on $[-1,1]$, their product on $[-1,1]^2$ is considered and transformed to the triangle $\mathbf{T}$ yielding the PKD polynomials

$$\phi_k(r,s) := g_{\ell m}(r,s) := P_\ell^{0,0}\left(\frac{2(1+r)}{(1-s)} - 1\right)\left(\frac{1-s}{2}\right)^\ell P_m^{2\ell+1,0}(s). \tag{9.26}$$

They form an orthogonal basis of $\mathbf{P}_N = \text{span}\{r^\ell s^m : 0 \leq \ell + m \leq N\}$ with respect to the scalar product

$$\langle \phi_i, \phi_j \rangle = \int_{\mathbf{T}^2} \phi_i(r,s)\phi_j(r,s)\mathrm{d}(r,s). \tag{9.27}$$

and fulfill the singular Sturm-Liouville-problem $\mathscr{L}_{r,s}\phi_k(r,s) + \lambda\phi_k(r,s) = 0$, where $(r,s) \in \mathbf{T}^2$ and

$$\begin{aligned}\mathscr{L}_{r,s} = \; & \tfrac{\partial}{\partial r}\left((1+r)\left((1-r)\tfrac{\partial}{\partial r} - (1+s)\tfrac{\partial}{\partial s}\right)\right) \\ & + \tfrac{\partial}{\partial s}\left((1+s)\left((1-s)\tfrac{\partial}{\partial s} - (1+r)\tfrac{\partial}{\partial r}\right)\right),\end{aligned}$$

which has been shown in [12].

This property is especially useful to construct a modal filter as in case of the original Spectral Viscosity method: We now solve the underlying conservation law with an additional viscosity term based on the differential operator $\mathscr{L}_{r,s}$, namely

$$u_t(\mathbf{x},t) + \nabla \cdot \mathscr{F}(u(\mathbf{x},t)) = \varepsilon(-1)^{p+1}\left(\mathscr{L}_{r,s}\right)^p u(\mathbf{x},t), \tag{9.28}$$

which can be done by a splitting approach in two steps. This results in filtering the coefficients $\hat{u}_k$ by the exponential filter

$$\sigma(\eta) = \mathrm{e}\left(-\frac{C_p n\,\Delta t}{h}\eta^{2p}\right) \tag{9.29}$$

where $\eta = \frac{k}{n}$, h is a measure for the length of the triangle and p is a chosen filter order with corresponding filter strength C_p. That means, the coefficients $\hat{u}_k$ are replaced by $\hat{u}_k^\sigma = \sigma\left(\frac{k}{n}\right)\hat{u}_k$. This modal filter has been investigated in [14] and exhibited very good behaviour for scalar testcases and the Euler equations. To apply

this filter only in those regions which are troubled by e.g. discontinuities, an efficient edge detector is needed, which brings us back to the application of the proposed Fourier based detector in the Spectral Difference method.

9.4.2 Application to the Spectral Difference Method

As mentioned above it would be useful to have a direct computation from the given modal coefficients to Fourier coefficients. Indeed, the following result can be proven for Proriol-Koornwinder-Dubiner polynomials.

Theorem 6. *If $\hat{u}_{\ell m}$ are the normed PKD coefficients, then*

$$\hat{\mathbf{f}}_{\xi\eta} = \sum_{\ell=0}^{n} \sum_{m=0}^{n-\ell} \hat{u}_{\ell m} \frac{\sqrt{\left(\ell+\frac{1}{2}\right)(\ell+m+1)}}{2^{\ell+2}} \sum_{j=0}^{\ell} \frac{(j+1)_\ell}{j!(\ell-j)!2^j} E(\xi, j) \cdot \sum_{k=0}^{m} \frac{(-1)^\ell (2\ell+k+2)_m}{k!(\ell-k)!2^k} E(\eta, k+\ell) \tag{9.30}$$

where $(x)_n := x(x+1)\cdots(x+n-1)$ *and* $E(\xi, j) = \int_{-1}^{1} (x-1)^j \mathrm{e}^{-\mathrm{i}\pi\xi x}\, \mathrm{d}x$.

This identity follows by inserting the reconstruction of u and the definitions of the PKD and Jacobi polynomials into the definition of $\hat{\mathbf{f}}_{\xi\eta}$, see [14]. The function E can be explicitly computed by partial integration and then reads

$$E(\xi, j) = \begin{cases} -\frac{(-2)^{j+1}}{j+1}, & \xi = 0, \\ \sum_{\mu=0}^{j-1} (j-\mu+1)_\mu \frac{(-1)^{\mu+1}(-2)^{j-\mu}}{(-\mathrm{i}\pi\xi)^{\mu+1}} + \frac{(-1)^j j!}{(-\mathrm{i}\pi\xi)^{j+1}} \left(\mathrm{e}^{-\mathrm{i}\pi\xi} - \mathrm{e}^{\mathrm{i}\pi\xi}\right), & \text{else.} \end{cases} \tag{9.31}$$

We now consider our nodal Spectral Difference method which uses modal filtering w.r.t. PKD basis on triangles as explained in Sect. 9.4.1. The algorithm to detect troubled cells is the following:

- Set the threshold value $\varepsilon_{\text{crit}}$ (global or local in critical regions, if a-priori known),
- For each triangle τ_i compute $\tilde{S}^K_{mn} f$ in $\mathring{\tau}_i$ (the inner part of a triangle),
- Mark cell as troubled if $\left|\tilde{S}^K_{mn} f\right| > \varepsilon_{\text{crit}}$.

An advantage is that one can estimate $\varepsilon_{\text{crit}}$ from the jump height since the conjugated sums converge pointwise to $d_{x_0}(f)$ resp. $d_{y_0}(f)$ resp. $d_{xy}(f)$, whereas in other edge detection approaches the critical parameter ε has to be chosen empirically. We will now present two testcases which show the abilities of the new edge detector.

Case 1. This testcase measures how many and which triangles the edge detector marks as troubled. To this end, we define a piecewise smooth function $f : [-1, 1]^2 \to \mathbf{R}$ by

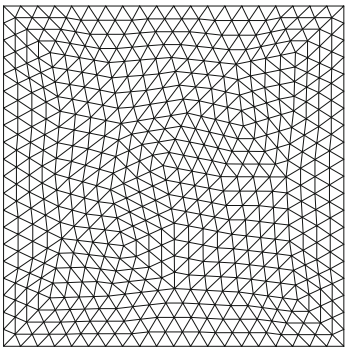


Fig. 9.7 Triangulation used in testcases 1 and 2

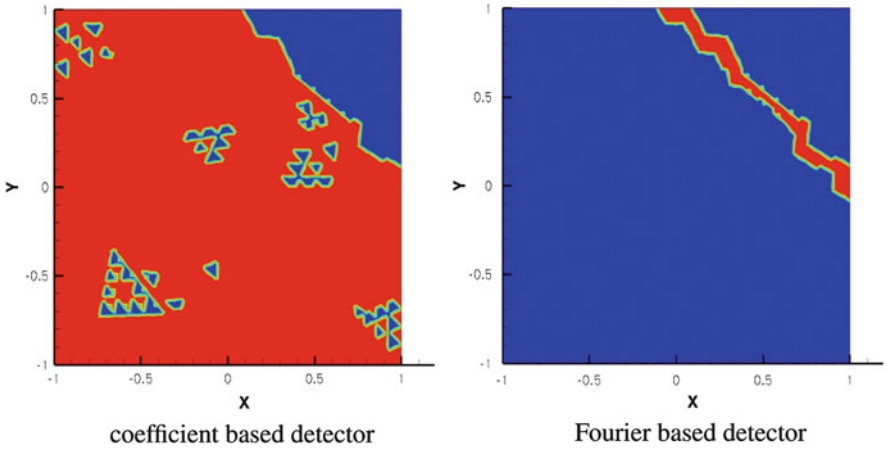


Fig. 9.8 Comparison of edge detectors; troubled *triangles* are *light-colored*

$$f(x,y) = \begin{cases} \sin(\pi(x+y)), & \text{if } x+y \leq 1, \\ 2, & \text{else,} \end{cases} \tag{9.32}$$

which has a discontinuity along $y = 1 - x$. Now lets consider numerical data given as in the third order Spectral Difference method with modal filtering for the grid with 1,088 triangles shown in Fig. 9.7, i.e. at 10 modal coefficients in each triangle. We will compare the proposed edge detector with a common shock indicator based on the decay of the leading coefficients given in [1]. Figure 9.8 shows the area which the edge detectors marks as troubled: the Fourier based one detects the discontinuity very sharply. In spite of this, the coefficient based detector marks too many cells and in partially wrong regions, namely the smooth sine part, as troubled. It is interesting that changing the parameter ε of the coefficient based detector did not improve the result noteworthy. Hence one can expect a better numerical solution from the Fourier

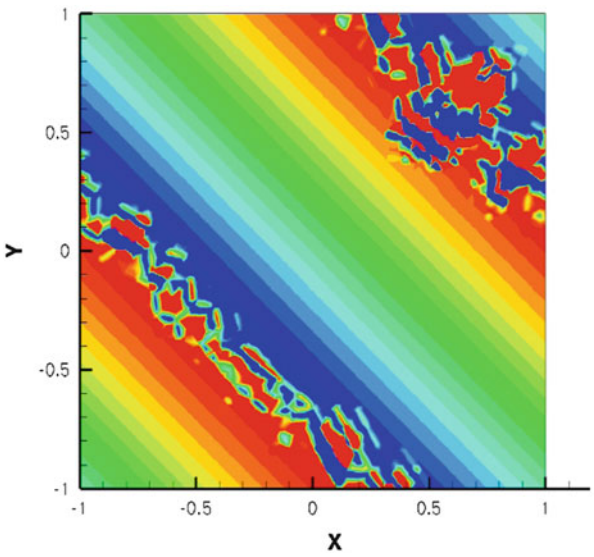


Fig. 9.9 Third order SD method for Burgers equation and sine initial condition without stabilization, $t = 0.45$. Same color legend as in Fig. 9.10

based detector since less cells are filtered (and consequently unnecessarily modified) due to the sharper detection.

Case 2. This testcase finally investigates how the Fourier based detector works in a concrete conservation law using the Spectral Difference method with the same parameters as in Case 1. We now consider the Burgers equation $u_t + uu_x + uu_y = 0$ in two space dimension with the smooth initial condition

$$u(x, y, 0) = u_0(x, y) = \sin(\pi(x + y)) \tag{9.33}$$

on the domain $[-1, 1]^2$. The exact solution develops two discontinuities at time $t = 0.45$ and stays smooth in between. Solving this equation without any additional stabilization leads to a breakdown of the numerical solution due to heavy oscillations around the arising discontinuities as can be seen in Fig. 9.9. Figure 9.10 shows the numerical solution which has been globally filtered up from time $t = 0.1$, i.e. without any shock indicator: the discontinuities are visible, but the numerical solution is way too much smeared due to the damping effect of the modal filters. Applying the coefficient based detector from [1] yields Fig. 9.11: the discontinuities are well captured, although the solution in the smooth part is too smeared in some areas. Figure 9.12 shows the same method with the same modal filter using the Fourier based detector: there is almost no difference in capturing the discontinuities, but the smooth part is now less influenced by the modal filter since the isolines in the sine part are more straight. This leads to a better approximation of the original solution.

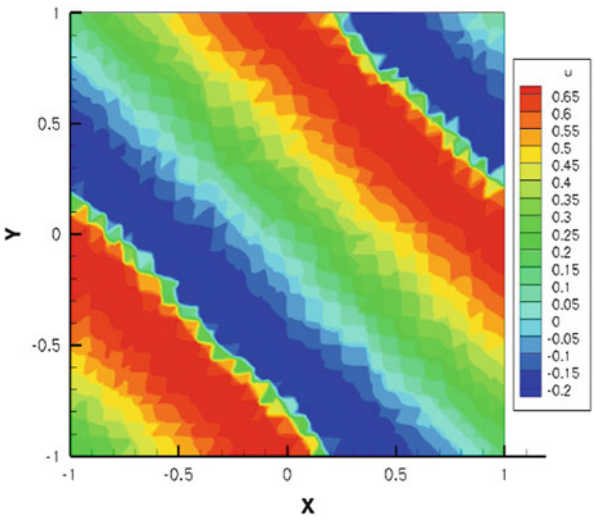


Fig. 9.10 Third order SD method for Burgers equation and sine initial condition with global filtering without any shock indicator, $t = 0.45$

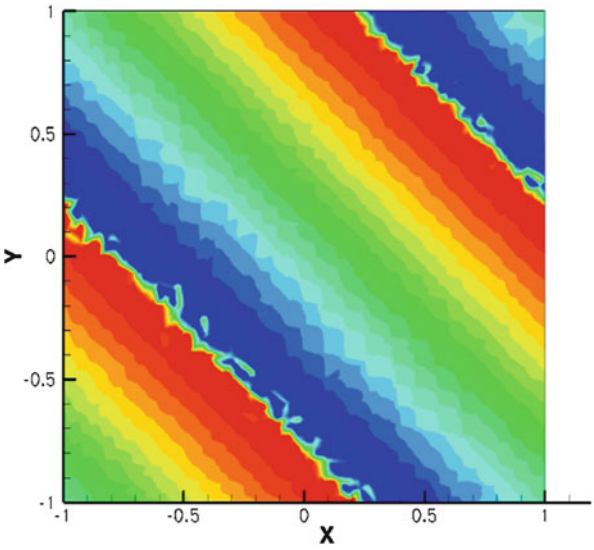


Fig. 9.11 Third order SD method for Burgers equation and sine initial condition with coefficient based detector, $t = 0.45$. Same color legend as in Fig. 9.10

9.5 Conclusion and Outlook

We have seen the extension of the generalized conjugated Fourier partial sums with admissible concentration kernels to the fully twodimensional case where the conjugated Fourier partial sums converge to the jump height in the mixed partial derivatives. Several examples with different concentration kernels were provided where the effect of the accelerated convergence is visible. Furthermore, the application of this edge detection approach in the context of the Spectral Difference

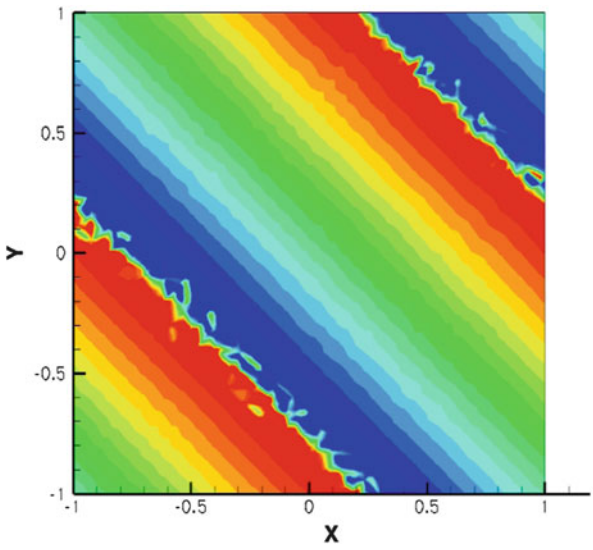


Fig. 9.12 Third order SD method for Burgers equation and sine initial condition with Fourier based detector, $t = 0.45$. Same color legend as in Fig. 9.10

method with modal filtering was enabled by the direct computation of Fourier from PKD coefficients. A comparison with a common coefficient based detector showed a better edge detection property of the Fourier based indicator both for initial conditions as well as for several time steps in the SD method. Another advantage of the new edge detector is the fact that the threshold $\varepsilon_{\text{crit}}$ can be better estimated since the oscillations are related to the jump height of the underlying function. If it is for example known that the smallest occurring discontinuities have height around 1, then $\frac{1}{2}$ would already be a good threshold.

Although the Fourier based detector proposes a superior approach to indicate shocks in local methods, it up to now lacks of computational efficiency due to the fact that the conjugated sum is computed in each triangle. One acceleration approach might be the computation of the conjugated partial sum in a greater domain from the given coefficients in each triangle. Here the question how to identify and implement an appropriate grid for the greater domain is subject to further research. Another point is to find similar connections between nodal and Fourier coefficients as in case of the PKD coefficients which is currently investigated. This would allow an implementation in nodal methods where no modal coefficients are used. Additionally, it would be interesting how this edge detector works in the presence of contact discontinuities or turbulences, and how it applies in different local methods as for example the Discontinuous Galerkin method.

Acknowledgements The author gratefully acknowledge partial financial support by the Deutsche For-schungsgemeinschaft (DFG) under grant SO 363/11-1.

References

1. Barter, G.E., Darmofal, D.L.: Shock-capturing with higher-order, PDE-based artificial viscosity. AIAA 2007-3823 (2007)
2. Blyth, M.G., Pozrikidis, C.: A Lobatto interpolation grid over the triangle. IMA J. Appl. Math. **71**, 153–169 (2006)
3. Dubiner, M.: Spectral methods on triangles and other domains. J. Sci. Comput. **6**, 345–390 (1991)
4. Gelb, A., Tadmor, E.: Detection of edges in spectral data. Appl. Comput. Harmon. Anal. **7**, 101–135 (1999)
5. Gelb, A., Tadmor, E.: Spectral reconstruction of one- and two-dimensional piecewise smooth functions from their discrete Data. Math. Model. Numer. Anal. **36**, 155–175 (2002)
6. Koornwinder, T.: Two-variable analogues of the classical orthogonal polynomials. In: Askey, R. (ed.) Theory and Applications of Special Functions. Academic, San Diego (1975)
7. Liu, Y., Vinokur, M., Wang, Z.J.: Spectral difference method for unstructured grids I: basic formulation. J. Comput. Phys. **216**, 780–801 (2006)
8. Lukács, F.: Ueber die Bestimmung des Sprunges einer Funktion aus ihrer Fourierreihe. Journal für die Reine und Angewandte Mathematik **150**, 107–112 (1920)
9. Meister, A., Ortleb, S., Sonar, Th., Wirz, M.: An extended discontinuous Galerkin and spectral difference method with modal filtering. ZAMM **6–7**, 459–464 (2013)
10. Móricz, F.: Extension of a theorem of Ferenc Lukács from single to double conjugate series. J. Math. Anal. Appl. **259**, 582–592 (2001)
11. Tadmor, E.: Super viscosity and spectral approximations of nonlinear conservation laws. Numer. Methods Fluid Dyn **4**, 69–82 (1993)
12. Taylor, M.A., Wingate, B.A.: The natural function space for triangular and tetrahedral spectral elements. Los Alamos National Laboratory report LA-UR-98-1711 (1998)
13. Wang, Z.J., Liu, Y., May, G., Jameson, A.: Spectral difference method for unstructured grids II: extension to the Euler equations. J. Sci. Comput. **32**, 45–71 (2007)
14. Wirz, M.: Ein Spektrale-Differenzen-Verfahren mit modaler Filterung und zweidimensionaler Kantendetektierung mithilfe konjugierter Fourierreihen. PhD thesis, Cuvillier Verlag Göttingen (2012)

Chapter 10
A Comparison of Analytical Solutions of a High-Order RBC Scheme and Its Equivalent Differential Equation for a Steady Shock Problem

Alain Lerat

Abstract The validity of the equivalent differential equation (also called modified equation or differential approximation) for representing shock solutions of high order schemes is investigated through a comparison of exact analytical solutions of the discrete scheme and its equivalent equation, for steady shocks of the inviscid Bürgers equation. For a third-order dissipative compact scheme, it is shown that the equivalent equation is a very good model of the scheme provided the dispersive term of fourth order is taken into account in addition to the dissipative term of third order.

10.1 Introduction

The development of high-order numerical schemes for solving nonlinear hyperbolic problems requires efficient tools to study and improve the shock capturing capabilities. Discrete shocks were first analysed in the 1970s by Jennings [10] for monotone schemes approximating a single conservation law and by Majda and Ralston [18] for dissipative first-order schemes applied to hyperbolic systems in the case of weak shocks. They proved the existence and stability of discrete shock profiles. Later the theory was further developed, notably by Michelson [19], Smyrlis [24], Serre [22], Arora and Roe [1], Bultelle, Grassin and Serre [3] and Benzoni-Gavage [2]. Recently the author [12] proposed an exact expression of steady discrete shocks for a class of residual-based compact (*RBC*) schemes of third, fifth and seventh-order [5,13,14] applied to the inviscid Bürgers equation. The aim of the present paper is to find a similar exact solution of the equivalent differential equation in order to study the validity of this equation for representing discrete shocks of high order schemes.

A. Lerat (✉)
DynFluid Laboratory, Arts et Métiers-ParisTech, 151 bd. de l'Hopital, 75013 Paris, France
e-mail: alain.lerat@ensam.eu

R. Abgrall et al. (eds.), *High Order Nonlinear Numerical Schemes for Evolutionary PDEs*, Lecture Notes in Computational Science and Engineering 99,
DOI 10.1007/978-3-319-05455-1_10,

Given a numerical scheme accurate at order p, its equivalent differential equation (also called modified equation or differential approximation) at order q ($q > p$) is a differential equation approximated by the scheme at order q. Considered as a model of the scheme, this equation has been successfully used for different purposes such as linear and nonlinear stability analysis [9,17,25,26] and control of parasitic oscillations [4,7,11,15], see also the text books [8,20,23]. However, the relevance of the approach for representing discrete shocks has not been established in a general situation. The most advanced work on this topic is due to Goodman and Majda [6] who proved the validity of the equivalent differential equation for a traveling shock of an upwind scheme approximating a scalar conservation law with the special nonlinear flux: $f(w) = -\log(\beta + \gamma\, \mathrm{e}^{-w})$, $\beta > 0$, $\gamma = 1 - \beta$. In the present paper, some more light is shed on this validity by considering a classical nonlinear flux and a higher-order scheme.

The paper is organized as follows. Section 10.2 briefly reminds the *RBC* schemes and Sect. 10.3 summarizes the construction of the exact solution of a steady discrete shock problem. Then Sect. 10.4 presents the equivalent differential equation of the *RBC* schemes. Such an equation can be exactly solved for the Bürgers equation and a steady shock profile. The solutions are given here for the equivalent equations at order 4 and 5 of the third order *RBC* scheme. These solutions are compared to the discrete shock solution for different locations of the shock in a mesh cell. Conclusions are drawn in Sect. 10.5.

10.2 Residual-Based Compact Schemes

For the one-dimensional hyperbolic system

$$\frac{\partial w}{\partial t} + \frac{\partial f}{\partial x} = 0 \tag{10.1}$$

where t is the time, x the space variable, w the state vector and $f = f(w)$ a smooth flux-function, a *RBC* scheme is a compact discrete form of

$$\frac{\partial w}{\partial t} + \frac{\partial f}{\partial x} = \frac{\delta x}{2}\frac{\partial}{\partial x}\left[\phi\,(\frac{\partial w}{\partial t} + \frac{\partial f}{\partial x})\right] \tag{10.2}$$

where ϕ is a numerical viscosity matrix depending only of the Jacobian-matrix $A = df/dw$ and δx is a spatial step. More precisely, consider a uniform mesh $x_j = j\,\delta x$ and introduce the classical difference and average operators:

$$(\delta v)_{j+\frac{1}{2}} = v_{j+1} - v_j \qquad (\mu v)_{j+\frac{1}{2}} = \frac{1}{2}(v_{j+1} + v_j)$$

where v_j is a mesh function, j being integer or half integer.

Table 10.1 *RBC* coefficients

	$\bar{a}$	$\bar{b}$	$\bar{c}$	a^{μ}	a^{δ}
RBC_3	0	$\frac{1}{6}$	0	0	0
RBC_5	$\frac{1}{10}$	$\frac{4}{15}$	$\frac{1}{90}$	$\frac{1}{12}$	$\frac{1}{6}$
RBC_7	$\frac{5}{42}$	$\frac{2}{7}$	$\frac{1}{70}$	$\frac{1}{10}$	$\frac{11}{60}$

The spatial approximation of (10.2) is

$$\tilde{r}_j = \frac{1}{2}[\delta(\phi\, \tilde{r}_1)]_j \tag{10.3}$$

where $\tilde{r}_j$ and $(\tilde{r}_1)_{j+\frac{1}{2}}$ are centered compact approximations of the exact residual $r = \partial w/\partial t + \partial f/\partial x$.

Limiting the scheme stencil to five points, these approximations can be written as:

$$\tilde{r}_j = \left[(I + \bar{b}\,\delta^2 + \bar{c}\,\delta^4)\frac{\partial w}{\partial t} + (I + \bar{a}\,\delta^2)\frac{\delta \mu f}{\delta x}\right]_j \tag{10.4}$$

$$(\tilde{r}_1)_{j+\frac{1}{2}} = \left[(I + a^{\mu}\,\delta^2)\,\mu\frac{\partial w}{\partial t} + (I + a^{\delta}\,\delta^2)\frac{\delta f}{\delta x}\right]_{j+\frac{1}{2}} \tag{10.5}$$

Using the coefficients given in Table 10.1, we obtain dissipative schemes spatially accurate at order 3, 5 and 7, denoted as RBC_3, RBC_5 and RBC_7 (see [5, 14]). Note that RBC_3 involves only three points.

The time approximation can also be made at various accuracy orders but does not matter in this paper devoted to the analysis of steady-state solutions. When such a solution is reached, the scheme (10.3) with (10.4) and (10.5) reduces to:

$$\left[(I + \bar{a}\,\delta^2)\frac{\delta \mu f}{\delta x}\right]_j = \frac{1}{2}\left[\delta\left(\phi\,(I + a^{\delta}\delta^2)\frac{\delta f}{\delta x}\right)\right]_j$$

which can also be written as:

$$\left(\delta f^*\right)_j = 0 \tag{10.6}$$

with the steady numerical flux

$$f^*_{j+\frac{1}{2}} = \left((I + \bar{a}\,\delta^2)\mu f - \frac{1}{2}\phi\,(I + a^{\delta}\delta^2)\delta f\right)_{j+\frac{1}{2}}. \tag{10.7}$$

Numerical viscosity matrices are described in [13, 16] for multidimensional *RBC* schemes. In the present one-dimensional case, the matrix ϕ reduces to the sign of the flux-Jacobian matrix:

$$\phi_{j+\frac{1}{2}} = \mathrm{sgn}A_{j+\frac{1}{2}} = \left[T_A\; Diag\left(\mathrm{sgn}\lambda_A^{(k)}\right) T_A^{-1}\right]_{j+\frac{1}{2}} \tag{10.8}$$

where T_A is a matrix the columns of which are the right eigenvectors of A and $Diag\left(\mathrm{sgn}\lambda_A^{(k)}\right)$ denotes a diagonal matrix having the sign of the eigenvalues of A as diagonal entries (the matrix A has real eigenvalues and a complete set of eigenvectors thanks to hyperbolicity of System (10.1)). The eigenvalues and eigenvectors of A are calculated at $x = (j + \frac{1}{2})\delta x$ using the Roe's average [21].

10.3 Steady Discrete Shocks

Consider the Bürgers equation, that is Eq. (10.1) for a scalar function w and the flux $f(w) = w^2/2$.

In one dimension, a steady shock is a stationary discontinuity between a constant left state w^L and a constant right state w^R satisfying the Rankine-Hugoniot condition

$$f(w^L) = f(w^R)$$

and the entropy condition $w^R < w^L$, that is

$$w^L > 0 \quad \text{and} \quad w^R = -w^L. \tag{10.9}$$

The location of the shock is not defined by the steady problem but depends on the unsteady evolution towards the steady state.

We study the approximation of a steady shock by the RBC_{2p-1} schemes ($p = 2, 3, 4$) in the space domain $[0, 1]$ discretized by $x_j = j\ \delta x, \quad j = 0, 1, \ldots, J$ with $\delta x = 1/J$. The discrete shock satisfies the steady-state equation (10.6) with the numerical flux $f^*_{j+\frac{1}{2}}$ given by (10.7), which gives:

$$f^*_{j+\frac{1}{2}} = C, \quad j = 0, 1, 2, \ldots, J-1 \tag{10.10}$$

where C is a constant. On the cell face $j + \frac{1}{2}$, the numerical flux involves the two points j and $j + 1$ for RBC_3 and the four points $j - 1, j, j + 1$ and $j + 2$ for RBC_5 and RBC_7. Thus for the latter, we need a special formula to compute the numerical flux at the cell face closest to a boundary point, i.e. for $j + \frac{1}{2} = \frac{1}{2}$ and $J - \frac{1}{2}$. On these two particular cell-faces, we choose to use the RBC_3 flux for the three schemes, that is:

$$f^*_{\frac{1}{2}} = \left(\mu f - \frac{1}{2}\phi\ \delta f\right)_{\frac{1}{2}} \qquad f^*_{J-\frac{1}{2}} = \left(\mu f - \frac{1}{2}\phi\ \delta f\right)_{J-\frac{1}{2}} \tag{10.11}$$

The boundary conditions (ingoing characteristic curves) are

$$w_0 = w^L \quad \text{and} \quad w_J = w^R$$

For the Bürgers equation, the matrix $\phi_{j+\frac{1}{2}}$ reduces to the scalar coefficient

$$\phi_{j+\frac{1}{2}} = \mathrm{sgn}(\mu w)_{j+\frac{1}{2}} = \begin{cases} +1 \text{ if } w_{j+1} + w_j \geq 0 \\ -1 \text{ if } w_{j+1} + w_j < 0 \end{cases}$$

We define an *integer* k such that

$$\phi_{j+\frac{1}{2}} = \begin{cases} +1 \text{ for } j = 0, 1, \ldots, k-1 \quad (\text{“supersonic zone”}) \\ -1 \text{ for } j = k, k+1, \ldots, J \quad (\text{“subsonic zone”}) \end{cases}$$

which means that the discrete-shock center is located between $x_{k-\frac{1}{2}}$ and $x_{k+\frac{1}{2}}$.

The constant C can be determined from the first numerical flux and the left boundary condition as:

$$C = f^*_{\frac{1}{2}} = \left(\mu f - \frac{1}{2}\delta f\right)_{\frac{1}{2}} = \frac{1}{2}(f_1 + f_0 - f_1 + f_0) = f_0 = f(w^L)$$

It is convenient to look for the discrete solution through the error on the flux at a mesh point:

$$e_j = f_j - f(w^L) \tag{10.12}$$

Since

$$(\delta f)_{j+\frac{1}{2}} = (\delta e)_{j+\frac{1}{2}} \qquad (\mu f)_{j+\frac{1}{2}} = f(w^L) + (\mu e)_{j+\frac{1}{2}}$$

the discrete problem becomes:

$$\left((I + \bar{a}\,\delta^2)\mu e - \frac{1}{2}\phi\,(I + a^{\delta}\delta^2)\delta e\right)_{j+\frac{1}{2}} = 0, \quad j = 1, 2, \ldots, J-2 \tag{10.13}$$

$$e_0 = 0 \qquad e_J = 0 \tag{10.14}$$

Note that we have only $J-2$ equations to determine the $J-1$ unknowns $e_1,\ e_2,\ \ldots, e_{J-1}$. This is related to the indeterminacy of the location of the steady shock in the cell $[x_{k-\frac{1}{2}},\ x_{k-\frac{1}{2}}]$. A way to remove it is to express the discrete solution in terms of the value e_k and relate this value to the shock location as in [12].

The discrete equation can be rewritten in the supersonic and subsonic zones as:

$$e_j + \alpha_1 \left(\delta^2 e\right)_{j+1} + \alpha_0 \left(\delta^2 e\right)_j = 0, \quad j = 1, 2, \ldots, k-1 \tag{10.15}$$

$$e_{j+1} + \alpha_0 \left(\delta^2 e\right)_{j+1} + \alpha_1 \left(\delta^2 e\right)_j = 0, \quad j = k, k+1, \ldots, J-2 \tag{10.16}$$

where

$$\alpha_0 = \frac{1}{2}\left(\bar{a} + a^{\delta}\right), \qquad \alpha_1 = \frac{1}{2}\left(\bar{a} - a^{\delta}\right)$$

For the scheme RBC_3, $\alpha_0 = \alpha_1 = 0$ and the discrete solution for the flux is simply:

$$e_j = 0, \qquad \text{for all } j \neq k \tag{10.17}$$

The value of e_k is free and does not affect the solution elsewhere.

Consider now the schemes RBC_5 and RBC_7 (for which α_0 and α_1 are not null) and look for solutions of the linear equation (10.15) in the form $e_j = K\beta^j$, which gives:

$$\left[1 + \alpha_1\left(\beta^2 - 2\beta + 1\right) + \alpha_0\left(\beta - 2 + \frac{1}{\beta}\right)\right] e_j = 0$$

Non-trivial solutions correspond to β-values canceling the above coefficient of e_j, that is

$$\beta^3 + a_2\beta^2 + a_1\beta + a_0 = 0 \tag{10.18}$$

where

$$a_2 = \frac{\alpha_0 - 2\alpha_1}{\alpha_1}, \quad a_1 = \frac{1 + \alpha_1 - 2\alpha_0}{\alpha_1}, \quad a_0 = \frac{\alpha_0}{\alpha_1}.$$

The polynomial roots can be conventionally obtained by reducing Eq. (10.18) to the form:

$$z^3 - p\,z - q = 0$$

where

$$z = \beta + \frac{a_2}{3}, \quad p = \frac{a_2^2}{3} - a_1, \quad q = \frac{a_2}{27}\left(9a_1 - 2a_2^2\right) - a_0.$$

The number of real roots depends on the sign of

$$\Delta = q^2 - \frac{4}{27}p^3.$$

For the schemes RBC_5 and RBC_7, Δ is negative so that the polynomial (10.18) has three real roots which can be written as (Cardano formula):

$$\beta_1 = b_0\cos(\theta) - \frac{a_2}{3}, \quad \beta_2 = b_0\cos(\theta + \frac{2\pi}{3}) - \frac{a_2}{3}, \quad \beta_3 = b_0\cos(\theta + \frac{4\pi}{3}) - \frac{a_2}{3},$$

Table 10.2 β-roots of the discrete schemes

	β_1	β_2	β_3
RBC_5	8.519917	−2.317318	−0.202600
RBC_7	9.049080	−2.097566	−0.247811

with

$$b_0 = 2\sqrt{\frac{p}{3}}, \quad \theta = \frac{1}{3}\arccos\left(\frac{q}{2}\sqrt{\frac{27}{p^3}}\right).$$

The values of the β-roots for RBC_5 and RBC_7 are given in Table 10.2. For both schemes, the first root β_1 is positive and the two others are negative.

The flux-solution of RBC_5 and RBC_7 in the supersonic zone is of the form:

$$e_j = \sum_{m=1}^{3} K_m \, \beta_m^j \tag{10.19}$$

where β_1, β_2, β_3 are the above real roots for each scheme and K_1, K_2 and K_3 are constant coefficients.

Similarly for Eq. (10.16) valid in the subsonic zone, we find that the β-roots are the inverse of those in the supersonic zone. Using the boundary conditions and the matching conditions between the two zones, we can determine the constants K_1, K_2 and K_3 in (10.19) and similar constants in the subsonic zone. For instance, by choosing the point k in the middle of the domain, i.e. $J = 2k$, we obtain the global flux-solution for RBC_5 and RBC_7 as:

$$e_j = \sum_{m=1}^{3} K_m \, \beta_m^{j'} \quad j' = \begin{cases} j & \text{for } j < k \\ J - j & \text{for } j > k \end{cases} \tag{10.20}$$

where the coefficients K_m are given by:

$$K_1 = e_k / b_1, \quad K_2 = b_2 K_1, \quad K_3 = b_3 K_1 \tag{10.21}$$

with

$$b_1 = \beta_1^k + b_2\beta_2^k + b_3\beta_3^k, \quad b_2 = \frac{b_{11} - b_{33}}{b_{33} - b_{22}}, \quad b_3 = \frac{b_{22} - b_{11}}{b_{33} - b_{22}}$$

where

$$b_{mm} = \beta_m^{k+1} - \beta_m^{k-1}, \; m = 1, 2, 3$$

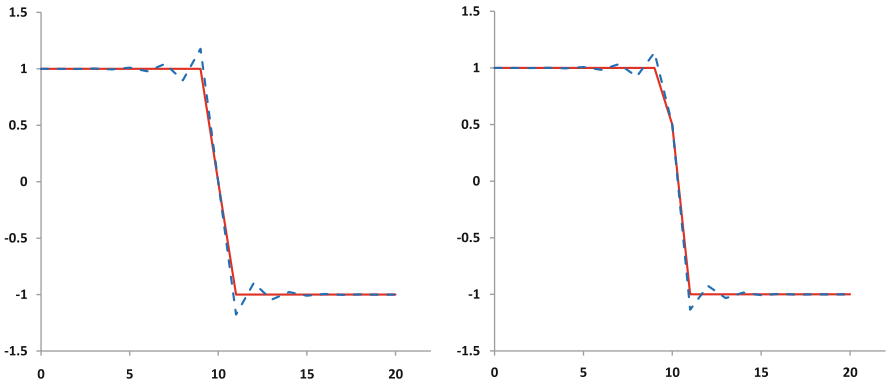


Fig. 10.1 RBC_3 solution (*solid line*) and RBC_7 solution (*dashed line*) for a steady shock at $X_s = \frac{x_s}{\delta x} = 10$ (*left side*) and 10.25 (*right side*)

The value of e_k is free, but contrary to the RBC_3 solution, it affects the whole shock profile through the coefficients (10.21).

The conservative variable is linked to the flux deviation e_j by:

$$w_j = \text{sgn}(k-j)\ \left[2(f(w^L) + e_j)\right]^{\frac{1}{2}} \qquad j = 0, 1, \ldots, J. \tag{10.22}$$

Using the above analytical results, it has been proved in [12] that:

- *RBC3* steady discrete shocks are always monotone. They are spread over two mesh cells in general and over one mesh interval if the shock is precisely located on a cell face ($x = (k + 1/2)\delta x$),
- *RBC5* and *RBC7* steady discrete shocks are oscillatory in general, but monotone and spread over one mesh interval if the shock is precisely located on a cell face,
- the oscillation amplitude of RBC5 and RBC7 is maximal when the shock is located on a mesh point (cell center) and it decreases monotonously as the shock location varies from a mesh point to a cell face.

Figures 10.1 and 10.2 show the theoretical discrete shocks of RBC_3 and RBC_7 obtained for $w^L = 1$ and $J = 2k = 20$, compared to the exact shocks, for three shock locations: $x = 10\delta x$ (shock on a mesh point), $x = 10.25\delta x$ and $x = 10.5\delta x$ (shock on a cell face). The RBC_5 shocks, slightly less oscillatory than the RBC_7 ones, have been omitted for clarity. Note that the theoretical discrete shocks are identical to the shocks directly computed by using the *RBC* schemes.

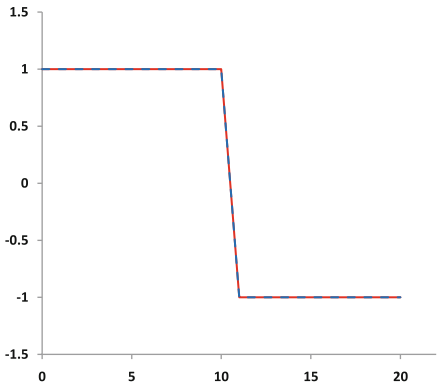


Fig. 10.2 RBC_3 solution (*solid line*) and RBC_7 solution (*dashed line*) for a steady shock at $X_s = 10.5$

Table 10.3 Dissipative and dispersive coefficients of the equivalent differential equation

	κ	γ
RBC_3	1/24	1/180
RBC_5	1/1,440	−1/9,540
RBC_7	1/5,600	1/44,100

10.4 Steady Shock Profiles of the Equivalent Differential Equation

For the hyperbolic system (10.1), we consider the RBC_{2p-1} schemes, $p = 2, 3, 4$ with a continuous time derivative (semi-discrete schemes). These schemes are spatially accurate at order $2p - 1$ and their equivalent differential equation at order $2p + 1$ is of the form:

$$\frac{\partial w}{\partial t} + \frac{\partial f}{\partial x} = (-1)^{p-1}(\delta x)^{2p-1}\ \kappa\ \frac{\partial}{\partial x}\left(\phi\ \frac{\partial^{2p-1} f}{\partial x^{2p-1}}\right) + (\delta x)^{2p}\ \gamma\ \frac{\partial^{2p+1} f}{\partial x^{2p+1}} \quad (10.23)$$

where $\kappa > 0$ and γ are scalar coefficients given in Table 10.3. The first error term is always dissipative and comes from the dissipative operator of the RBC_{2p-1} scheme, i.e. the right-hand side of (10.3), see [16]. The second error term is dispersive and comes from the left-hand side of (10.3).

Let us study the capability of the equivalent equation (10.23) to represent a shock solution of the scheme. For a shock wave moving at speed s, consider a coordinate frame traveling with the shock:

$$X = \frac{x - s\,t}{\delta x} \qquad T = t$$

The time and space derivatives become

$$\frac{\partial w}{\partial t} = \frac{\partial w}{\partial T} - \frac{s}{\delta x}\frac{\partial w}{\partial X} \qquad \frac{\partial f}{\partial x} = \frac{1}{\delta x}\frac{\partial f}{\partial X}$$

Assuming the shock has a constant profile, we have $\partial w/\partial T = 0$ and the equivalent equation becomes:

$$\frac{\mathrm{d}}{\mathrm{d}X}(F - s\,W) = (-1)^{p-1}\,\kappa\,\frac{\mathrm{d}}{\mathrm{d}X}\left(\phi\,\frac{\mathrm{d}^{2p-1}F}{\mathrm{d}X^{2p-1}}\right) + \gamma\,\frac{\mathrm{d}^{2p+1}F}{\mathrm{d}X^{2p+1}}$$

where $W(X) = w(x,t)$ and $F(X) = f[W(X)]$.

Consider now the scalar case with the Bürgers flux ($f(w) = w^2/2$) and a steady shock ($s = 0$). Then the above ordinary differential equation (ODE) can be integrated analytically in a bounded space domain $0 < X < X_1$, splitted into a "supersonic" subdomain ($W(X) > 0$, i.e. $\phi = 1$) and a "subsonic" subdomain ($W(X) < 0$, i.e. $\phi = -1$). Consider for simplicity the RBC3 scheme ($p = 2$). In the supersonic zone, the steady shock profile is governed by the linear ODE:

$$\frac{\mathrm{d}F}{\mathrm{d}X} = -\kappa\,\frac{\mathrm{d}^4F}{\mathrm{d}X^4} + \gamma\,\frac{\mathrm{d}^5F}{\mathrm{d}X^5}, \qquad 0 < X < X_s \tag{10.24}$$

where $X = X_s$ is the shock center defined by $W(X_s) = 0$. The boundary conditions are

$$F(0) = f^L, \quad F(X_s) = 0, \quad \frac{\mathrm{d}F}{\mathrm{d}X}(0) = 0, \quad \frac{\mathrm{d}^2F}{\mathrm{d}X^2}(0) = 0, \quad \frac{\mathrm{d}^3F}{\mathrm{d}X^3}(0) = 0 \tag{10.25}$$

Integrating once Eq. (10.24), we obtain:

$$F = -\kappa\,\frac{\mathrm{d}^3F}{\mathrm{d}X^3} + \gamma\,\frac{\mathrm{d}^4F}{\mathrm{d}X^4} + C_0, \qquad 0 < X < X_s \tag{10.26}$$

where C_0 is a constant. Setting

$$E(X) = F(X) - C_0 \tag{10.27}$$

and $a_3 = -\kappa/\gamma = -15/2$ and $a_0 = -1/\gamma = -180$, Eq. (10.26) can be rewritten as:

$$\frac{\mathrm{d}^4E}{\mathrm{d}X^4} + a_3\frac{\mathrm{d}^3E}{\mathrm{d}X^3} + a_0E = 0, \qquad 0 < X < X_s \tag{10.28}$$

with the boundary conditions

$$E(0) - E(X_s) = f^L, \quad \frac{\mathrm{d}E}{\mathrm{d}X}(0) = 0, \quad \frac{\mathrm{d}^2E}{\mathrm{d}X^2}(0) = 0, \quad \frac{\mathrm{d}^3E}{\mathrm{d}X^3}(0) = 0 \tag{10.29}$$

Looking for solutions of (10.28) in the form $E = K\,\exp(\beta X)$, we obtain the characteristic equation

$$\beta^4 + a_3\beta^3 + a_0 = 0$$

This quartic equation can be solved by radicals using the Ferrari approach (a particular case of the Galois theory). We find two real roots and two conjugate complex roots:

$$\beta_1 < 0, \quad \beta_2 > 0, \quad \beta_3 = \nu - \mathrm{i}\theta, \quad \beta_4 = \nu + \mathrm{i}\theta$$

where $\beta_1 = -2.61110662$, $\beta_2 = 7.86936328$, $\nu = 1.12087167$, $\theta = 2.73929396$.
Thus in the supersonic zone, the general solution of (10.28) is

$$E(X) = C_1 \exp(\beta_1 X) + C_2 \exp(\beta_2 X) + [C_3 \cos(\theta X) + C_4 \sin(\theta X)] \exp(\nu X) \tag{10.30}$$

Calculating E and its derivatives at $X = 0$ and also $E(X_s)$, we obtain the real constants C_1, C_2, C_3 and C_4 through the solution of a linear algebraic system expressing the boundary conditions (10.29) for a given value of the shock center X_s. Then from (10.27), we deduce the constant C_0:

$$C_0 = F(0) - E(0) = f^L - (C_1 + C_2 + C_3)$$

The conservative variable in the supersonic zone is given by:

$$W(X) = [2(E(X) + C_0)]^{\frac{1}{2}}, \quad 0 < X < X_s \tag{10.31}$$

In the subsonic zone, the steady shock profile is governed by:

$$\frac{\mathrm{d}F}{\mathrm{d}X} = \kappa \frac{\mathrm{d}^4 F}{\mathrm{d}X^4} + \gamma \frac{\mathrm{d}^5 F}{\mathrm{d}X^5}, \quad X_s < X < X_1 \tag{10.32}$$

with the boundary conditions:

$$F(X_s) = 0, \ F(X_1) = f^R = f^L, \ \frac{\mathrm{d}F}{\mathrm{d}X}(X_1) = 0, \ \frac{\mathrm{d}^2 F}{\mathrm{d}X^2}(X_1) = 0, \ \frac{\mathrm{d}^3 F}{\mathrm{d}X^3}(X_1) = 0 \tag{10.33}$$

Using the new variable $X' = X_1 - X$, the ODE (10.32) takes the same form as (10.24). Therefore the subsonic branch is given by $E(X')$ where the function E is defined by (10.30). With the variable X', the boundary conditions (10.33) take the form (10.25), except the condition $F(X_s) = 0$ which becomes $F(X_1 - X_s) = 0$. Thus, the constants C_m, $m = 0, 1, \ldots, 4$ are modified, except if $X_s = X_1/2$. With new constants C'_m, the conservative variable is:

$$W(X) = -\left[2(E(X_1 - X) + C'_0)\right]^{\frac{1}{2}}, \quad X_s < X < X_1 \tag{10.34}$$

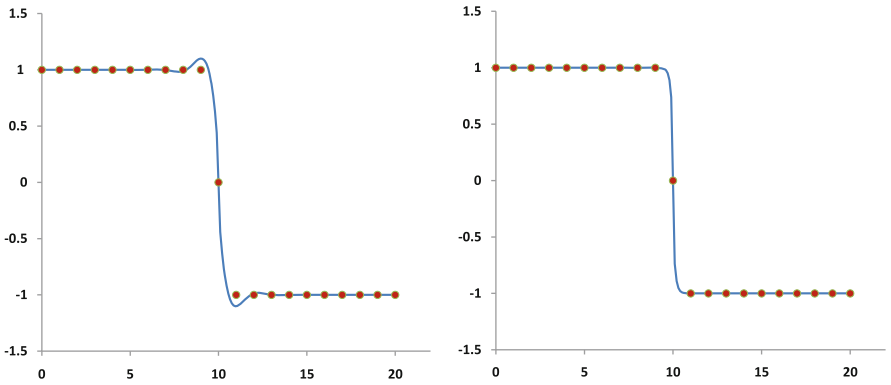


Fig. 10.3 Solution of the differential equation at order 4 (*left side*) and order 5 (*right side*) compared to the RBC_3 discrete-solution (*solid symbols*) for a steady shock at $X = 10$

It is interesting to study what happens without the dispersive term in the differential equation, that is to consider the equivalent differential equation at order 4 (rather than 5) for RBC_3. In the supersonic zone, Eqs. (10.24) with $\gamma = 0$ becomes:

$$\frac{\mathrm{d}F}{\mathrm{d}X} = -\kappa \frac{\mathrm{d}^4 F}{\mathrm{d}X^4}, \quad 0 < X < X_s \tag{10.35}$$

and (10.28) reduces to:

$$\frac{\mathrm{d}^3 E}{\mathrm{d}X^3} + b_0 E = 0, \quad 0 < X < X_s \tag{10.36}$$

with $b_0 = 1/\kappa = 24$ and the boundary conditions

$$E(0) - E(X_s) = f^L, \quad \frac{\mathrm{d}E}{\mathrm{d}X}(0) = 0, \quad \frac{\mathrm{d}^2 E}{\mathrm{d}X^2}(0) = 0 \tag{10.37}$$

The characteristic equation reduces to

$$\beta^3 + b_0 = 0$$

and has the obvious roots:

$$\beta_1 < 0, \quad \beta_2 = \nu - \mathrm{i}\theta, \quad \beta_3 = \nu + \mathrm{i}\theta$$

where $\beta_1 = -2(3^{\frac{1}{3}})$, $\nu = 3^{\frac{1}{3}}$, $\theta = 3^{\frac{1}{2}} 3^{\frac{1}{3}}$.

Thus, in the supersonic zone, the general solution of (10.36) is

$$E(X) = C_1 \exp(\beta_1 X) + [C_2 \cos(\theta X) + C_3 \sin(\theta X)] \exp(\nu X) \tag{10.38}$$

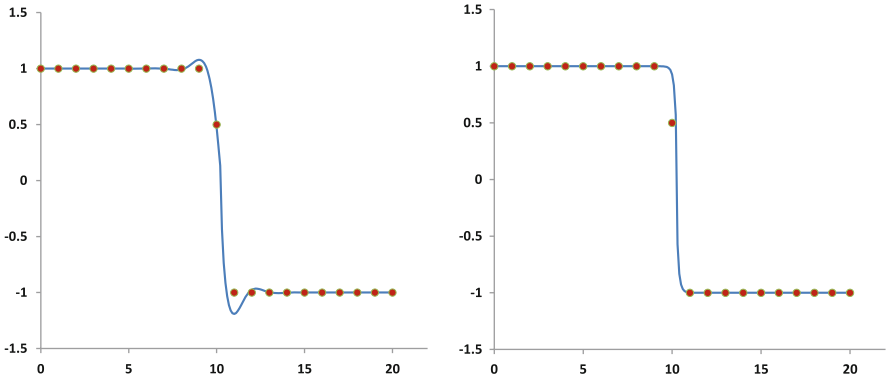


Fig. 10.4 Solution of the differential equation at order 4 (*left side*) and order 5 (*right side*) compared to the RBC_3 discrete-solution (*solid symbols*) for a steady shock at $X = 10.25$

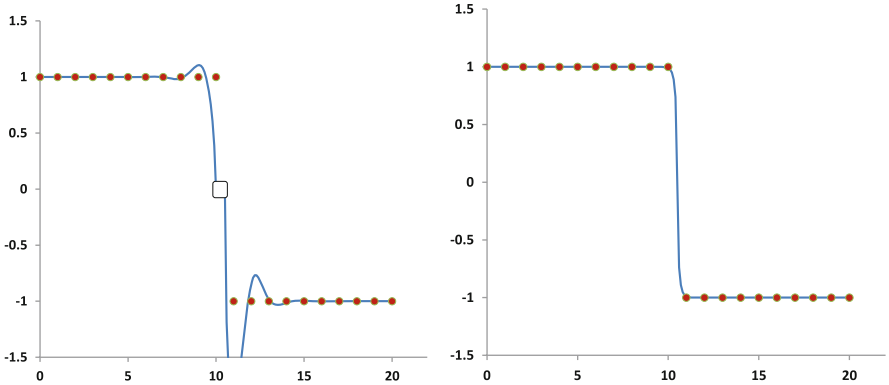


Fig. 10.5 Solution of the differential equation at order 4 (*left side*) and order 5 (*right side*) compared to the RBC_3 discrete-solution (*solid symbols*) for a steady shock at $X = 10.50$

where C_1, C_2 and C_3 are determined from (10.37). The conservative variable is given by (10.31) with $C_0 = f^L - (C_1 + C_2)$. Again, the analogue of (10.38) in the subsonic zone reads $E(X_1 - X)$ with different C-constants.

The above shock solutions of the equivalent differential equations at order 4 and 5 of the RBC_3 scheme are compared to the discrete shock on Figs. 10.3–10.5 corresponding to the three shock locations $X = x/\delta x = 10$ (shock on a mesh point), $X = 10.25$ and $X = 10.5$ (shock on a cell face). These results show that the use of the equivalent differential equation at order 4 is not sufficient to represent correctly the discrete shocks. Furthermore when the discrete shock is very steep (shock on a cell face), the shock profile of the equivalent differential equation at order 4 *does not exist*, since in the vicinity of the "sonic point", there is a small region ($10 \leq X < 10.5$) in which the flux takes negative values. On the contrary, the equivalent differential equation at order 5 mimics remarkably well the discrete scheme.

10.5 Conclusions

The validity of the equivalent differential equation for analysing shock capturing properties of residual-based compact schemes has been studied through a comparison of exact analytical solutions of the discrete scheme and its equivalent equation, for steady shocks of the inviscid Bürgers equation. For the third-order *RBC* scheme, it has been shown that the equivalent equation at order 4 (Bürgers equation plus a dissipative term of order δx^3) does not represent properly the discrete shock profile, specially when it is very stiff. On the contrary, the equivalent equation at order 5 (including a dispersive term of order δx^4) appears as an excellent model of the scheme for shock solutions. Similar studies are in progress for higher order *RBC* schemes.

References

1. Arora, M., Roe, P.L.: On postshock oscillations due to shock capturing schemes in unsteady flows. J. Comput. Phys. **130**, 25–40 (1997)
2. Benzoni-Gavage, S.: Semi-discrete shock profiles for hyperbolic systems of conservation laws. Physica D **115**, 109–123 (1998)
3. Bultelle, M., Grassin, M., Serre, D.: Unstable Godunov discrete profiles for steady shock waves. SIAM J. Numer. Anal. **35**, 2272–2297 (1998)
4. Chin, R.: Dispersion and Gibbs phenomenon associated with difference approximations to initial boundary-value problems for hyperbolic equations. J. Comput. Phys. **18**, 233–247 (1975)
5. Corre, C., Falissard, F., Lerat, A.: High-order residual-based compact schemes for compressible inviscid flows. Comput. Fluids **36**, 1567–1582 (2007)
6. Goodman, J., Majda, A.: The validity of the modified equation for nonlinear shock waves. J. Comput. Phys. **58**, 336–348 (1985)
7. Hedstrom, G.W.: Models of difference schemes for $u_t + u_x = 0$ by partial differential equations. Math. Comput. **29**, 969–977 (1975)
8. Hirsch, C.: Numerical Computation of Internal and External Flows: Fundamentals of Numerical Discretization. Wiley, Chichester/New York/Brisbane/Toronto/Singapore (1988)
9. Hirt, C.W.: Heuristic stability theory for finite difference equtions. J. Comput. Phys. **2**, 339–355 (1968)
10. Jennings, G.: Discrete shocks. Commun. Pure Appl. Math. **27**, 25–37 (1974)
11. Lerat, A.: Numerical shock structure and nonlinear corrections for difference schemes in conservation form. Lect. Notes Phys. **90**, 345–351 (1979)
12. Lerat, A.: Steady discrete shocks of high-order RBC schemes. J. Comput. Phys. **252**, 350–364 (2013)
13. Lerat, A., Corre, C.: A residual-based compact scheme for the compressible Navier-Stokes equations. J. Comput. Phys. **170**, 642–675 (2001)
14. Lerat, A., Corre, C.: High order residual-based compact schemes on structured grids. In: CFD-Higher Order Discretization Methods, pp. 1–111. VKI LS 2006-1, Rhode Saint Genese (2006)
15. Lerat, A., Peyret, R.: The problem of spurious oscillations in the numerical solution of the equations of gas dynamics. Lect. Notes Phys. **35**, 251–256 (1975)
16. Lerat, A., Grimich, K., Cinnella, P.: On the design of high order residual-based dissipation for unsteady compressible flows. J. Comput. Phys. **235**, 32–51 (2013)

17. Majda, A., Osher, S.: A systematic approach for correcting nonlinear instabilities. Numerische Mathematik **30**, 429–452 (1978)
18. Majda, A., Ralston, J.: Discrete shock profiles for systems of conservation laws. Commun. Pure Appl. Math. **32**, 445–482 (1979)
19. Michelson, D.: Discrete shocks profiles for difference approximations to systems of conservation laws. Adv. Appl. Math. **5**, 433–469 (1984)
20. Peyret, R., Taylor,T.D.: Computational Methods for Fluid Flow. Springer, New York (1983)
21. Roe, P.L.: Approximate Riemann solvers, parameter vectors and difference schemes. J. Comput. Phys. **43**, 357–372 (1981)
22. Serre, D.: Remarks about the discrete profiles of shock waves. Matemática Contemporânea **11**, 153–170 (1996)
23. Shokin, Y.I.: The Method of Differential Approximation. Springer, Berlin/New York (1983)
24. Smyrlis, Y.S.: Existence and stability of stationary profiles of the LW scheme. Commun. Pure Appl. Math. **43**, 509–545 (1990)
25. Warming, R.F., Hyett, B.J.: The modified equation approach to the stability and accuracy analysis of finite-difference methods. J. Comput. Phys. **14**, 159–179 (1974)
26. Yanenko, N.N., Shokin, Y.I.: First differential approximation method and approximate viscosity of difference schemes. Phys. Fluids **12**(Suppl. II), 28–33 (1969)

Chapter 11
Discontinuous Galerkin Method and Applications to Fluid-Structure Interaction Problems

Miloslav Feistauer, Jan Česenek, Martin Hadrava, Jaromír Horáček, and Adam Kosík

Abstract The subject of the paper is the numerical simulation of viscous compressible flow in time dependent domains. The motion of the boundary of the domain occupied by the fluid is taken into account with the aid of the ALE (Arbitrary Lagrangian-Eulerian) formulation of the Navier-Stokes equations. The flow problem is coupled with the dynamical linear elasticity problem. Both problems are discretized in space by the discontinuous Galerkin (DG) finite element method using piecewise polynomial discontinuous approximations. The time discretization is carried out by the BDF scheme or the DG in time. The developed methods are tested by numerical experiments and applied to the solution of a fluid-structure interaction problem.

M. Feistauer (✉)
Faculty of Mathematics and Physics, Department of Numerical Mathematics, Charles University in Prague, Sokolovská 83, 18675 Praha 8, Czech Republic

Institute of Thermomechanics, The Academy of Sciences of the Czech Republic, v.v.i., Dolejškova 1402/5, 182 00 Praha 8, Czech Republic
e-mail: feist@karlin.mff.cuni.cz

J. Česenek
Aerospace Research and Test Establishment, Beranových 130, 199 05 Praha-Letňany, Czech Republic
e-mail: jan.cessa@seznam.cz

M. Hadrava • A. Kosík
Faculty of Mathematics and Physics, Department of Numerical Mathematics, Charles University in Prague, Sokolovská 83, 18675 Praha 8, Czech Republic
e-mail: martin@hadrava.eu; adam.kosik@atlas.cz

J. Horáček
Institute of Thermomechanics, The Academy of Sciences of the Czech Republic, v.v.i., Dolejškova 1402/5, 182 00 Praha 8, Czech Republic
e-mail: jaromirh@it.cas.cz

R. Abgrall et al. (eds.), *High Order Nonlinear Numerical Schemes for Evolutionary PDEs*, Lecture Notes in Computational Science and Engineering 99,
DOI 10.1007/978-3-319-05455-1_11,

11.1 Introduction

In a number of practical applications, flow problems in time dependent domains and fluid-structure interaction (FSI) have to be solved. The flow-induced vibrations of elastic structures may affect negatively the operation and stability of the systems. Therefore, one of the main goals of aeroelasticity is the prediction and alleviation of the aeroelastic instability. This discipline achieved many results, particularly from the engineering point of view (see, e.g. the monographs [3, 8] and [16]).

Most of works concerned with FSI problems assume that the flowing medium is incompressible. Nevertheless, there is a number of flow problems, where the compressibility plays an important role. In the numerical solution of compressible flow, it is necessary to overcome several obstacles. In high-speed flow shock waves and contact discontinuities appear and (in viscous flow) it is necessary to resolve boundary layers and wakes. The solution of low Mach number flow is also rather difficult. This is caused by the stiff behaviour of numerical schemes and acoustic phenomena appearing in low Mach number flow at incompressible limit.

It appears that a suitable numerical method for the solution of compressible flow overcoming these difficulties is the discontinuous Galerkin method (DGM). It employs piecewise polynomial approximations without any requirement on the continuity on interfaces between neighbouring elements. The DGM uses advantages of the finite volume method and the finite element method. It was applied to the numerical simulation of the compressible Euler equations, for example, by Bassi and Rebay in [1], where the space DG discretization was combined with explicit Runge-Kutta time discretization. In [2] Baumann and Oden apply an hp version for the solution of the space DG discretization with explicit time stepping to compressible flow. Van der Vegt and van der Ven use the space-time discontinuous Galerkin method for the solution of the Euler equations in [19], where the discrete problem is solved with the aid of a multigrid accelerated pseudo-time-integration.

In our paper we shall describe a numerical method based on the application of the DGM for the solution of compressible flow which is coupled with the solution of systems describing the deformations and vibrations of elastic structures induced by aerodynamical forces. The compressible Navier-Stokes equations describing the compressible flow as well as the dynamical elasticity equations are discretized in space by the discontinuous Galerkin method. For the time discretization the BDF method or DG in time are used. The resulting method is applied to the solution of the interaction of viscous compressible flow with an elastic airfoil. The paper contains results of numerical experiments showing the accuracy and robustness of the developed numerical techniques.

11.2 Continuous Problem

In this section we shall formulate the continuous flow and structural problems and their interaction.

11.2.1 Flow Problem

We consider compressible flow in a bounded domain $\Omega_t \subset \mathbb{R}^2$ depending on time $t \in [0, T]$. The system describing compressible flow, consisting of the continuity equation, the Navier-Stokes equations and the energy equation, can be written in the form [12]

$$\frac{\partial \boldsymbol{w}}{\partial t} + \sum_{s=1}^{2} \frac{\partial \boldsymbol{f}_s(\boldsymbol{w})}{\partial x_s} = \sum_{s=1}^{2} \frac{\partial \boldsymbol{R}_s(\boldsymbol{w}, \nabla \boldsymbol{w})}{\partial x_s}, \tag{11.1}$$

where

$$\begin{aligned}
\boldsymbol{w} &= (w_1, \dots, w_4)^T = (\rho, \rho v_1, \rho v_2, E)^T \in \mathbb{R}^4, \qquad (11.2)\\
\boldsymbol{w} &= \boldsymbol{w}(x,t),\ x \in \Omega_t,\ t \in (0,T),\\
\boldsymbol{f}_i(\boldsymbol{w}) &= (f_{i1}, \cdots, f_{i4})^T = (\rho v_i, \rho v_1 v_i + \delta_{1i}\, p, \rho v_2 v_i + \delta_{2i}\, p, (E+p)v_i)^T,\\
\boldsymbol{R}_i(\boldsymbol{w}, \nabla \boldsymbol{w}) &= (R_{i1}, \dots, R_{i4})^T = \left(0, \tau_{i1}^V, \tau_{i2}^V, \tau_{i1}^V\, v_1 + \tau_{i2}^V\, v_2 + k\frac{\partial \theta}{\partial x_i}\right)^T,\\
\tau_{ij}^V &= \lambda\, \mathrm{div}\boldsymbol{v}\, \delta_{ij} + 2\mu\, d_{ij}(\boldsymbol{v}),\ d_{ij}(\boldsymbol{v}) = \frac{1}{2}\left(\frac{\partial v_i}{\partial x_j} + \frac{\partial v_j}{\partial x_i}\right).
\end{aligned}$$

We use the following notation: ρ – fluid density, p – pressure, E – total energy, $\boldsymbol{v} = (v_1, v_2)$ – velocity, θ – absolute temperature, $\gamma > 1$ – Poisson adiabatic constant, $c_v > 0$ – specific heat at constant volume, $\mu > 0, \lambda = -2\mu/3$ – viscosity coefficients, k – heat conduction. The vector-valued function $\boldsymbol{w}$ is called the state vector, the functions $\boldsymbol{f}_i$ are the so-called inviscid fluxes and $\boldsymbol{R}_i$ represent viscous terms.

The above system is completed by the thermodynamical relations

$$p = (\gamma - 1)(E - \frac{1}{2}\rho|\boldsymbol{v}|^2), \quad \theta = \frac{1}{c_v}\Big(\frac{E}{\rho} - \frac{1}{2}|\boldsymbol{v}|^2\Big). \tag{11.3}$$

The resulting system is equipped with the initial condition

$$\boldsymbol{w}(x,0) = \boldsymbol{w}^0(x), \quad x \in \Omega_0, \tag{11.4}$$

and boundary conditions. We assume that the boundary of Ω_t consists of three disjoint parts: $\partial\Omega_t = \Gamma_I \cup \Gamma_O \cup \Gamma_{Wt}$, where Γ_I is the inlet, Γ_O is the outlet and Γ_{Wt} denotes impermeable walls that may move in dependence on time. On the individual parts of the boundary we prescribe the following conditions:

$$\text{(a)}\ \rho|_{\Gamma_I} = \rho_D, \quad \text{(b)}\ \boldsymbol{v}|_{\Gamma_I} = \boldsymbol{v}_D = (v_{D1}, v_{D2})^{\mathrm{T}}, \tag{11.5}$$

(c) $\sum_{i,j=1}^{2} \tau_{ij}^V n_i v_j + k\frac{\partial \theta}{\partial n} = 0 \quad \text{on } \Gamma_I,$

(d) $\boldsymbol{v}|_{\Gamma_{Wt}} = \bar{\boldsymbol{z}}_D =$ velocity of a moving wall, (e) $\frac{\partial \theta}{\partial n}|_{\Gamma_{Wt}} = 0$ on Γ_{Wt},

(f) $\sum_{i=1}^{2} \tau_{ij}^V n_i = 0, \quad j = 1, 2,$ (g) $\frac{\partial \theta}{\partial n} = 0$ on Γ_O.

It is easy to see that $\boldsymbol{f}_s(\alpha \boldsymbol{w}) = \alpha \boldsymbol{f}_s(\boldsymbol{w})$ for $\alpha > 0$. This implies that

$$\boldsymbol{f}_s(\boldsymbol{w}) = \mathbb{A}_s(\boldsymbol{w})\boldsymbol{w}, \quad s = 1, 2, \tag{11.6}$$

where $\mathbb{A}_s(\boldsymbol{w}) = D\boldsymbol{f}_s(\boldsymbol{w})/D\boldsymbol{w}$, $s = 1, 2$, are the Jacobi matrices of the mappings $\boldsymbol{f}_s$. The viscous terms $\boldsymbol{R}_s(\boldsymbol{w}, \nabla \boldsymbol{w})$ can be expressed in the form

$$\boldsymbol{R}_s(\boldsymbol{w}, \nabla \boldsymbol{w}) = \sum_{k=1}^{2} \mathbb{K}_{s,k}(\boldsymbol{w}) \frac{\partial \boldsymbol{w}}{\partial x_k}, \quad s = 1, 2, \tag{11.7}$$

where $\mathbb{K}_{s,k}(\boldsymbol{w}) \in \mathbb{R}^{4\times 4}$ are matrices depending on $\boldsymbol{w}$ (cf. [6]).

In order to take into account the time dependence of the domain, we use the so-called *arbitrary Lagrangian-Eulerian* (ALE) technique, see e.g. [17]. It is based on a regular one-to-one ALE mapping of the reference domain Ω_0 onto Ω_t:

$$\mathscr{A}_t : \overline{\Omega}_0 \longrightarrow \overline{\Omega}_t, \text{ i.e., } \boldsymbol{X} \in \overline{\Omega}_0 \longmapsto \boldsymbol{x} = \boldsymbol{x}(\boldsymbol{X}, t) = \mathscr{A}_t(\boldsymbol{X}) \in \overline{\Omega}_t.$$

Here we use the notation $\boldsymbol{X}$ for points in $\overline{\Omega}_0$ and $\boldsymbol{x}$ for points in $\overline{\Omega}_t$.

Further, we define the domain velocity:

$$\tilde{\boldsymbol{z}}(\boldsymbol{X}, t) = \frac{\partial}{\partial t}\mathscr{A}_t(\boldsymbol{X}), \quad t \in [0, T], \ \boldsymbol{X} \in \Omega_0,$$
$$\bar{\boldsymbol{z}}(\boldsymbol{x}, t) = \tilde{\boldsymbol{z}}(\mathscr{A}^{-1}(\boldsymbol{x}), t), \quad t \in [0, T], \ \boldsymbol{x} \in \Omega_t,$$

$\bar{\boldsymbol{z}} = (\bar{z}_1, \bar{z}_2)$, and the ALE derivative of a function $f = f(\boldsymbol{x}, t)$ defined for $\boldsymbol{x} \in \Omega_t$ and $t \in [0, T]$:

$$\frac{D^A}{Dt} f(\boldsymbol{x}, t) = \frac{\partial \tilde{f}}{\partial t}(\boldsymbol{X}, t), \tag{11.8}$$

where

$$\tilde{f}(\boldsymbol{X}, t) = f(\mathscr{A}_t(\boldsymbol{X}), t), \ \boldsymbol{X} \in \Omega_0, \ \boldsymbol{x} = \mathscr{A}_t(\boldsymbol{X}).$$

By the chain rule,

$$\frac{D^A f}{Dt} = \frac{\partial f}{\partial t} + \operatorname{div}(\bar{z} f) - f \operatorname{div} \bar{z}.$$

This leads to the ALE formulation of the Navier-Stokes equations

$$\frac{D^A \boldsymbol{w}}{Dt} + \sum_{s=1}^{2} \frac{\partial \boldsymbol{g}_s(\boldsymbol{w})}{\partial x_s} + \boldsymbol{w} \operatorname{div} \bar{z} = \sum_{s=1}^{2} \frac{\partial \boldsymbol{R}_s(\boldsymbol{w}, \nabla \boldsymbol{w})}{\partial x_s}, \tag{11.9}$$

where

$$\boldsymbol{g}_s(\boldsymbol{w}) = \boldsymbol{f}_s(\boldsymbol{w}) - \bar{z}_s \boldsymbol{w}, \quad s = 1, 2,$$

are the ALE modified inviscid fluxes.

We see that in the ALE formulation of the Navier-Stokes equations the time derivative $\partial \boldsymbol{w}/\partial t$ is replaced by the ALE derivative $D^A \boldsymbol{w}/Dt$, the inviscid fluxes $\boldsymbol{f}_s$ are replaced by the ALE modified inviscid fluxes $\boldsymbol{g}_s$ and a new additional "reaction" term $\boldsymbol{w}$ div$z$ appears.

System (11.9) is completed by the thermodynamical relations (11.3), the initial conditions (11.4) and the boundary conditions (11.5).

11.2.2 Dynamical Elasticity Problem

We consider an elastic body represented by a bounded open set $\Omega^b \subset \mathbb{R}^2$ with the boundary formed by two disjoint parts Γ_D^b and $\Gamma_N^b \subset \Gamma_{W0}$. The set Γ_N^b is a common part of the boundary $\partial\Omega^b$ with the boundary the reference domain Ω_0 occupied by the fluid at time $t = 0$. See Fig. 11.5. We denote by $\boldsymbol{u}(\boldsymbol{X}, t) = (u_1(\boldsymbol{X}, t), u_2(\boldsymbol{X}, t))$, $\boldsymbol{X} = (X_1, X_2) \in \Omega^b$, $t \in (0, T)$, the displacement of the body. The gradient of the function $\boldsymbol{u}$, denoted by $\nabla \boldsymbol{u}$, is defined by $\nabla \boldsymbol{u} = \left(\partial u_i / \partial X_j\right)_{i,j=1}^2$.

The dynamical elasticity problem is defined as follows: given $T > 0$ and a domain $\Omega^b \subset \mathbb{R}^2$ with boundary $\partial\Omega^b = \Gamma_D^b \cup \Gamma_N^b$, we seek for a displacement function $\boldsymbol{u} : \Omega^b \times [0, T] \to \mathbb{R}^2$ such that

$$\rho^b \frac{\partial^2 \boldsymbol{u}}{\partial t^2} + c_M \rho^b \frac{\partial \boldsymbol{u}}{\partial t} - \operatorname{div} \boldsymbol{\sigma}(\boldsymbol{u}) - c_K \frac{\partial}{\partial t} \operatorname{div} \boldsymbol{\sigma}(\boldsymbol{u}) = \boldsymbol{f} \quad \text{in } \Omega^b \times (0, T), \tag{11.10}$$

$$\boldsymbol{u} = \boldsymbol{u}_D \quad \text{in } \Gamma_D^b \times (0, T), \tag{11.11}$$

$$\boldsymbol{\sigma}(\boldsymbol{u}) \cdot \boldsymbol{n} = \boldsymbol{g}_N \quad \text{in } \Gamma_N^b \times (0, T), \tag{11.12}$$

$$\boldsymbol{u}(x, 0) = \boldsymbol{u}_0(x), \quad x \in \Omega^b, \tag{11.13}$$

$$\frac{\partial \boldsymbol{u}}{\partial t}(x, 0) = \boldsymbol{z}_0(x), \quad x \in \Omega^b. \tag{11.14}$$

Here $\boldsymbol{f} : \Omega^b \times (0,T) \to \mathbb{R}^2$ (outer volume force), $\boldsymbol{u}_D : \Gamma_D^b \times (0,T) \to \mathbb{R}^2$ (boundary displacement), $\boldsymbol{g}_N : \Gamma_N^b \times (0,T) \to \mathbb{R}^2$ (boundary normal stress), $\boldsymbol{u}_0 : \Omega^b \to \mathbb{R}^2$ (initial displacement), $\boldsymbol{z}_0 : \Omega^b \to \mathbb{R}^2$ (initial deformation velocity) and $\rho^b > 0$ (material density) are given functions. Moreover,

$$\boldsymbol{\sigma}(\boldsymbol{u}) = \lambda^b \operatorname{tr}(\boldsymbol{e}(\boldsymbol{u}))\mathbb{I} + 2\mu^b \boldsymbol{e}(\boldsymbol{u}), \quad \boldsymbol{e}(\boldsymbol{u}) = \frac{1}{2}\left(\nabla \boldsymbol{u} + \nabla \boldsymbol{u}^T\right), \tag{11.15}$$

where $\boldsymbol{\sigma} = (\sigma_{ij})_{i,j=1}^2$ is the stress tensor, $\boldsymbol{e} = (e_{ij})_{i,j=1}^2$ is the strain tensor, $\mathbb{I}$ is the unit tensor, $\operatorname{tr}(\boldsymbol{e}) = \sum_{i=1}^2 e_{ii} = \operatorname{div}\boldsymbol{u}$ and λ^b and μ^b are the Lamé parameters. They are related to the Young modulus E^b and the Poisson ratio σ^b by

$$\lambda^b = \frac{E^b \sigma^b}{(1+\sigma^b)(1-2\sigma^b)}, \quad \mu^b = \frac{E^b}{2(1+\sigma^b)}. \tag{11.16}$$

The expressions $c_M \rho^b \frac{\partial \boldsymbol{u}}{\partial t}$ and $c_K \frac{\partial}{\partial t} \operatorname{div} \boldsymbol{\sigma}(\boldsymbol{u})$ represent the damping terms, with c_M, $c_K \geq 0$.

11.2.3 Interaction of the Compressible Flow and Dynamical Elasticity Problem

Up to now the fluid flow and the deformation of the elastic body have been considered as two separate problems. Now we need to take into account the mutual interaction of the fluid and the body (see Fig. 11.1) on the common boundary

$$\tilde{\Gamma}_{Wt} = \left\{\boldsymbol{x} \in \mathbb{R}^2;\ \boldsymbol{x} = \boldsymbol{X} + \boldsymbol{u}(\boldsymbol{X}, t),\ \boldsymbol{X} \in \Gamma_N^b\right\}.$$

The domain Ω_t is determined by the displacement $\boldsymbol{u}$ on Γ_N^b at time t. If the domain Ω_t occupied by the fluid at time t is known, we can solve the problem describing the flow and compute the surface force acting on the body on $\tilde{\Gamma}_{Wt}$, which can be transformed to the reference configuration, i.e., to the interface Γ_N^b. In the case of the linear elasticity model, when only small deformations are considered, we get the transmission condition

$$\sum_{j=1}^2 \tau_{ij}^b(\boldsymbol{X}) n_j(\boldsymbol{X}) = -\sum_{j=1}^2 \tau_{ij}^f(\boldsymbol{x}) n_j(\boldsymbol{X}), \quad i = 1,2, \tag{11.17}$$

where τ_{ij}^f are the components of the stress tensor of the fluid: $\tau_{ij}^f = -p\delta_{ij} + \tau_{ij}^V$, $i, j = 1, 2$. The points $\boldsymbol{x}$ and $\boldsymbol{X}$ satisfy the relation

$$\boldsymbol{x} = \boldsymbol{X} + \boldsymbol{u}(\boldsymbol{X}, t) \tag{11.18}$$

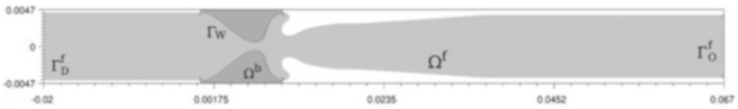


Fig. 11.1 The domain occupied by the fluid and the parts of the boundary formed by elastic bodies

and $\boldsymbol{n}(\boldsymbol{X}) = (n_1(\boldsymbol{X}), n_2(\boldsymbol{X}))$ denotes the unit outer normal to the body Ω^b on Γ_N^b at the point $\boldsymbol{X}$. Further, the fluid velocity is defined on the moving part of the boundary Γ_{Wt} by the transmission condition

$$\boldsymbol{v}(\boldsymbol{x}, t) = \bar{\boldsymbol{z}}_D(\boldsymbol{x}, t) = \frac{\partial \boldsymbol{u}(\boldsymbol{X}, t)}{\partial t}. \tag{11.19}$$

The ALE mapping $\mathscr{A}_t$ is determined with the aid of an artificial stationary elasticity problem, where we seek $\boldsymbol{d} = (d_1, d_2)$ defined in Ω_0 as a solution of the elastic system

$$\sum_{j=1}^{2} \frac{\partial \tau_{ij}^a}{\partial x_j} = 0 \text{ in } \Omega_0, \quad i = 1, 2, \tag{11.20}$$

where τ_{ij}^a are the components of the artificial stress tensor $\tau_{ij}^a = \lambda^a \mathrm{div}\boldsymbol{d}\,\delta_{i,j} + 2\mu^a e_{ij}^a(\boldsymbol{d})$, $e_{ij}^a(\boldsymbol{d}) = \frac{1}{2}\left(\frac{\partial d_i}{\partial x_j} + \frac{\partial d_j}{\partial x_i}\right)$, $i = 1, 2$. The Lamé coefficients λ^a and μ^a are related to the artificial Young modulus E^a and the artificial Poisson number σ^a similarly as in Sect. 11.2.2. The boundary conditions for $\boldsymbol{d}$ are prescribed by

$$\boldsymbol{d}|_{\Gamma_I \cup \Gamma_O} = 0, \ \boldsymbol{d}|_{\Gamma_{W0} \setminus \Gamma_N^b} = 0, \ \boldsymbol{d}(\boldsymbol{x}, t) = \boldsymbol{u}(\boldsymbol{x}, t), \ \boldsymbol{x} \in \Gamma_N^b. \tag{11.21}$$

The solution of the problem (11.20)–(11.21) gives us the ALE mapping of $\overline{\Omega}_0$ onto $\overline{\Omega}_t$ in the form

$$\mathscr{A}_t(\boldsymbol{x}) = \boldsymbol{x} + \boldsymbol{d}(\boldsymbol{x}, t), \quad \boldsymbol{x} \in \overline{\Omega}_0, \tag{11.22}$$

for each time instant t.

11.3 Discrete Problem

11.3.1 Space Discretization of the Flow Problem by the Discontinuous Galerkin Method

For the space semidiscretization of the flow problem we use the discontinuous Galerkin method (DGM). We construct a polygonal approximation Ω_{ht} of the domain Ω_t. By $\mathscr{T}_{ht}$ we denote a partition of the closure $\overline{\Omega}_{ht}$ of the domain Ω_{ht}

into a finite number of closed triangles K with mutually disjoint interiors such that $\overline{\Omega}_{ht} = \bigcup_{K \in \mathscr{T}_{ht}} K$.

By $\mathscr{F}_{ht}$ we denote the system of all faces of all elements $K \in \mathscr{T}_{ht}$. Further, we introduce the set of boundary faces $\mathscr{F}_{ht}^B = \{\Gamma \in \mathscr{F}_{ht};\ \Gamma \subset \partial\Omega_{ht}\}$, the set of "Dirichlet" boundary faces $\mathscr{F}_{ht}^D = \{\Gamma \in \mathscr{F}_{ht}^B;$ a Dirichlet condition is prescribed on $\Gamma\}$ and the set of inner faces $\mathscr{F}_{ht}^I = \mathscr{F}_{ht} \setminus \mathscr{F}_{ht}^B$. Each $\Gamma \in \mathscr{F}_{ht}$ is associated with a unit normal vector $\boldsymbol{n}_\Gamma$ to Γ. For $\Gamma \in \mathscr{F}_{ht}^B$ the normal $\boldsymbol{n}_\Gamma$ has the same orientation as the outer normal to $\partial\Omega_{ht}$. We set $h_\Gamma =$ length of $\Gamma \in \mathscr{F}_{ht}$ and $h_K =$ diameter of $K \in \mathscr{T}_{ht}$.

For each $\Gamma \in \mathscr{F}_{ht}^I$ there exist two neighbouring elements $K_\Gamma^{(L)}, K_\Gamma^{(R)} \in \mathscr{T}_h$ such that $\Gamma \subset \partial K_\Gamma^{(R)} \cap \partial K_\Gamma^{(L)}$. We use the convention that $K_\Gamma^{(R)}$ lies in the direction of $\boldsymbol{n}_\Gamma$ and $K_\Gamma^{(L)}$ lies in the opposite direction to $\boldsymbol{n}_\Gamma$. If $\Gamma \in \mathscr{F}_{ht}^B$, then the element adjacent to Γ will be denoted by $K_\Gamma^{(L)}$.

Now we introduce the space of piecewise polynomial functions

$$\boldsymbol{S}_{ht}^r = [S_{ht}^r]^4, \quad \text{with } S_{ht}^r = \{v; v|_K \in P_r(K)\ \forall\, K \in \mathscr{T}_{ht}\}, \tag{11.23}$$

where $r > 0$ is an integer and $P_r(K)$ denotes the space of all polynomials on K of degree $\le r$. A function $\boldsymbol{\varphi} \in \boldsymbol{S}_{ht}^r$ is, in general, discontinuous on interfaces $\Gamma \in \mathscr{F}_{ht}^I$. By $\boldsymbol{\varphi}_\Gamma^{(L)}$ and $\boldsymbol{\varphi}_\Gamma^{(R)}$ we denote the values of $\boldsymbol{\varphi}$ on Γ considered from the interior and the exterior of $K_\Gamma^{(L)}$, respectively, and set $\langle \boldsymbol{\varphi} \rangle_\Gamma = (\boldsymbol{\varphi}_\Gamma^{(L)} + \boldsymbol{\varphi}_\Gamma^{(R)})/2$, $[\boldsymbol{\varphi}]_\Gamma = \boldsymbol{\varphi}_\Gamma^{(L)} - \boldsymbol{\varphi}_\Gamma^{(R)}$.

The discrete problem is derived in the following way: We multiply system (11.9) by a test function $\boldsymbol{\varphi}_h \in \boldsymbol{S}_{ht}^r$, integrate over $K \in \mathscr{T}_{ht}$, apply Green's theorem, sum over all elements $K \in \mathscr{T}_{ht}$, use the concept of the numerical flux and introduce suitable terms mutually vanishing for a regular exact solution and linearize the resulting forms on the basis of properties (11.6) and (11.7) of the functions $\boldsymbol{f}_s$ and $\boldsymbol{R}_s$ (see, e.g. [4, 14]). In this way we get the following forms (followed by the explanation of symbols appearing in their definitions):

$$\begin{aligned}
\hat{a}_h(\overline{\boldsymbol{w}}_h, \boldsymbol{w}_h, \boldsymbol{\varphi}_h, t) = &\sum_{K \in \mathscr{T}_{ht}} \int_K \sum_{s=1}^{2} \sum_{k=1}^{2} \mathbb{K}_{s,k}(\overline{\boldsymbol{w}}_h)\, \frac{\partial \boldsymbol{w}_h}{\partial x_k} \cdot \frac{\partial \boldsymbol{\varphi}_h}{\partial x_s}\, \mathrm{d}\boldsymbol{x} \\
&- \sum_{\Gamma \in \mathscr{F}_{ht}^I} \int_\Gamma \sum_{s=1}^{2} \left\langle \sum_{k=1}^{2} \mathbb{K}_{s,k}(\overline{\boldsymbol{w}}_h) \frac{\partial \boldsymbol{w}_h}{\partial x_k} \right\rangle (\boldsymbol{n}_\Gamma)_s \cdot [\boldsymbol{\varphi}_h]\, \mathrm{d}S \\
&- \sum_{\Gamma \in \mathscr{F}_{ht}^D} \int_\Gamma \sum_{s=1}^{2} \sum_{k=1}^{2} \mathbb{K}_{s,k}(\overline{\boldsymbol{w}}_h) \frac{\partial \boldsymbol{w}_h}{\partial x_k} (\boldsymbol{n}_\Gamma)_s \cdot \boldsymbol{\varphi}_h\, \mathrm{d}S \\
&- \Theta \sum_{\Gamma \in \mathscr{F}_{ht}^I} \int_\Gamma \sum_{s=1}^{2} \left\langle \sum_{k=1}^{2} \mathbb{K}_{k,s}^T(\overline{\boldsymbol{w}}_h) \frac{\partial \boldsymbol{\varphi}_h}{\partial x_k} \right\rangle (\boldsymbol{n}_\Gamma)_s \cdot [\boldsymbol{w}_h]\, \mathrm{d}S
\end{aligned} \tag{11.24}$$

$$- \Theta \sum_{\Gamma \in \mathscr{F}_{ht}^D} \int_\Gamma \sum_{s=1}^{2} \sum_{k=1}^{2} \mathbb{K}_{k,s}^T(\overline{\boldsymbol{w}}_h) \frac{\partial \boldsymbol{\varphi}_h}{\partial x_k} (\boldsymbol{n}_\Gamma)_s \cdot \boldsymbol{w}_h \, \mathrm{d}S,$$

$$d_h(\boldsymbol{w}_h, \boldsymbol{\varphi}_h, t) = \sum_{K \in \mathscr{T}_{ht}} \int_K (\boldsymbol{w}_h \cdot \boldsymbol{\varphi}_h) \operatorname{div} \overline{\boldsymbol{z}} \, \mathrm{d}x, \tag{11.25}$$

$$J_h(\boldsymbol{w}_h, \boldsymbol{\varphi}_h, t) = \sum_{\Gamma \in \mathscr{F}_{ht}^I} \int_\Gamma \frac{\mu C_W}{h_\Gamma} [\boldsymbol{w}_h] \cdot [\boldsymbol{\varphi}_h] \, \mathrm{d}S + \sum_{\Gamma \in \mathscr{F}_{ht}^D} \int_\Gamma \frac{\mu C_W}{h_\Gamma} \boldsymbol{w}_h \cdot \boldsymbol{\varphi}_h \, \mathrm{d}S, \tag{11.26}$$

$$\ell_h(\boldsymbol{w}_h, \boldsymbol{\varphi}_h, t) = \sum_{\Gamma \in \mathscr{F}_{ht}^D} \int_\Gamma \frac{\mu C_W}{h_\Gamma} \boldsymbol{w}_B \cdot \boldsymbol{\varphi}_h \, \mathrm{d}S \tag{11.27}$$

$$- \Theta \sum_{\Gamma \in \mathscr{F}_{ht}^D} \int_\Gamma \sum_{s=1}^{2} \sum_{k=1}^{2} \mathbb{K}_{k,s}^T(\overline{\boldsymbol{w}}_h) \frac{\partial \boldsymbol{\varphi}_h}{\partial x_k} (\boldsymbol{n}_\Gamma)_s \cdot \boldsymbol{w}_B \, \mathrm{d}S,$$

$$\hat{b}_h(\overline{\boldsymbol{w}}_h, \boldsymbol{w}_h, \boldsymbol{\varphi}_h, t) = \tag{11.28}$$

$$- \sum_{K \in \mathscr{T}_{ht}} \int_K \sum_{s=1}^{2} ((\mathbb{A}_s(\overline{\boldsymbol{w}}_h(x)) - \overline{z}_s(x)) \mathbb{I} \boldsymbol{w}_h(x)) \cdot \frac{\partial \boldsymbol{\varphi}_h(x)}{\partial x_s} dx$$

$$+ \sum_{\Gamma \in \mathscr{F}_{ht}^I} \int_\Gamma \left(\mathbb{P}_g^+ \left(\langle \overline{\boldsymbol{w}}_h \rangle_\Gamma, \boldsymbol{n}_\Gamma \right) \boldsymbol{w}_h^{(L)} + \mathbb{P}_g^- \left(\langle \overline{\boldsymbol{w}}_h \rangle_\Gamma, \boldsymbol{n}_\Gamma \right) \boldsymbol{w}_h^{(R)} \right) \cdot [\boldsymbol{\varphi}_h] \, \mathrm{d}S$$

$$+ \sum_{\Gamma \in \mathscr{F}_{ht}^B} \int_\Gamma \left(\mathbb{P}_g^+ \left(\langle \overline{\boldsymbol{w}}_h \rangle_\Gamma, \boldsymbol{n}_\Gamma \right) \boldsymbol{w}_h^{(L)} + \mathbb{P}_g^- \left(\langle \overline{\boldsymbol{w}}_h \rangle_\Gamma, \boldsymbol{n}_\Gamma \right) \overline{\boldsymbol{w}}_h^{(R)} \right) \cdot \boldsymbol{\varphi}_h \, \mathrm{d}S,$$

We set $\Theta = 1$ or $\Theta = 0$ or $\Theta = -1$ and get the so-called symmetric version (SIPG) or incomplete version (IIPG) or nonsymmetric version (NIPG), respectively, of the discretization of viscous terms. In (11.26) and (11.27) C_W denotes a positive sufficiently large constant.

The symbols $\mathbb{P}_g^+(\boldsymbol{w}, \boldsymbol{n})$ and $\mathbb{P}_g^-(\boldsymbol{w}, \boldsymbol{n})$ denote the "positive" and "negative" parts of the matrix $\mathbb{P}_g(\boldsymbol{w}, \boldsymbol{n}) = \sum_{s=1}^{2} (\mathbb{A}_s(\boldsymbol{w}) - \overline{z}_s \mathbb{I}) n_s$ defined in the following way. By [10,12], this matrix is diagonalizable. It means that there exists a nonsingular matrix $\mathbb{T} = \mathbb{T}(\boldsymbol{w}, \boldsymbol{n})$ such that

$$\mathbb{P}_g = \mathbb{T} I\!\Lambda \mathbb{T}^{-1}, \quad I\!\Lambda = \operatorname{diag}(\lambda_1, \ldots, \lambda_4), \tag{11.29}$$

where $\lambda_i = \lambda_i(\boldsymbol{w}, \boldsymbol{n})$ are eigenvalues of the matrix $\mathbb{P}_g$. Now we define the "positive" and "negative" parts of the matrix $\mathbb{P}_g$ by

$$\mathbb{P}_g^{\pm} = \mathbb{T} I\Lambda^{\pm} \mathbb{T}^{-1}, \quad I\Lambda^{\pm} = \mathrm{diag}(\lambda_1^{\pm}, \dots, \lambda_4^{\pm}), \tag{11.30}$$

where $\lambda^+ = \max(\lambda, 0)$, $\lambda^- = \min(\lambda, 0)$. It is possible to show that the integrals along edges Γ were obtained by the linearization of an ALE modified Vijayasundaram numerical flux (cf. [20] or [12])

$$\boldsymbol{H}(\boldsymbol{w}_L, \boldsymbol{w}_R, \boldsymbol{n}) = \mathbb{P}_g^+\Big(\frac{\boldsymbol{w}_L + \boldsymbol{w}_R}{2}, \boldsymbol{n}\Big)\boldsymbol{w}_L + \mathbb{P}_g^-\Big(\frac{\boldsymbol{w}_L + \boldsymbol{w}_R}{2}, \boldsymbol{n}\Big)\boldsymbol{w}_R. \tag{11.31}$$

The boundary state $\boldsymbol{w}_B$ is defined on the basis of the Dirichlet boundary conditions (11.5), (a), (b), (d) and extrapolation:

$$\boldsymbol{w}_B = (\rho_D, \rho_D v_{D1}, \rho_D v_{D2}, c_v \rho_D \theta_\Gamma^{(L)} + \frac{1}{2}\rho_D |\boldsymbol{v}_D|^2) \quad \text{on } \Gamma_I, \tag{11.32}$$

$$\boldsymbol{w}_B = \boldsymbol{w}_\Gamma^{(L)} \quad \text{on } \Gamma_O, \tag{11.33}$$

$$\boldsymbol{w}_B = (\rho_\Gamma^{(L)}, \rho_\Gamma^{(L)} \overline{z}_{D1}, \rho_\Gamma^{(L)} \overline{z}_{D2}, c_v \rho_\Gamma^{(L)} \theta_\Gamma^{(L)} + \frac{1}{2}\rho_\Gamma^{(L)} |\overline{z}_D|^2) \quad \text{on } \Gamma_{Wt}. \tag{11.34}$$

For $\Gamma \in \mathscr{F}_{ht}^B$ we set $\langle \overline{\boldsymbol{w}}_h \rangle_\Gamma = (\overline{\boldsymbol{w}}_\Gamma^{(L)} + \overline{\boldsymbol{w}}_\Gamma^{(R)})/2$ and the boundary state $\overline{\boldsymbol{w}}_\Gamma^{(R)}$ is defined with the aid of the solution of the 1D linearized initial-boundary Riemann problem as in [13].

In order to avoid spurious oscillations in the approximate solution in the vicinity of discontinuities or steep gradients, we apply artificial viscosity forms. They are based on the discontinuity indicator

$$g_t(K) = \int_{\partial K} [\overline{\rho}_h]^2 \, \mathrm{d}S \big/ (h_K |K|^{3/4}), \quad K \in \mathscr{T}_{ht}, \tag{11.35}$$

introduced in [7]. By $[\overline{\rho}_h]$ we denote the jump of the function $\overline{\rho}_h$ on the boundary ∂K and $|K|$ denotes the area of the element K. Then we define the discrete discontinuity indicator $G_t(K) = 0$ if $g_t(K) < 1$, $G_t(K) = 1$ if $g_t(K) \geq 1$, and the artificial viscosity forms (see [11])

$$\hat{\beta}_h(\overline{\boldsymbol{w}}_h, \boldsymbol{w}_h, \boldsymbol{\varphi}_h, t) = \nu_1 \sum_{K \in \mathscr{T}_{ht}} h_K G_t(K) \int_K \nabla \boldsymbol{w}_h \cdot \nabla \boldsymbol{\varphi}_h \, \mathrm{d}\boldsymbol{x}, \tag{11.36}$$

$$\hat{J}_h(\overline{\boldsymbol{w}}_h, \boldsymbol{w}_h, \boldsymbol{\varphi}_h, t) = \frac{1}{2}\nu_2 \sum_{\Gamma \in \mathscr{F}_{ht}^I} \big(G_t(K_\Gamma^{(L)}) + G_t(K_\Gamma^{(R)})\big) \int_\Gamma [\boldsymbol{w}_h] \cdot [\boldsymbol{\varphi}_h] \, \mathrm{d}S,$$

with parameters $\nu_1, \nu_2 = O(1)$.

11.3.2 Time Discretization of the Flow Problem

In what follows, we shall describe two possible techniques for the time discretization of the flow problem.

11.3.2.1 Time Discretization of Flow Problem by the BDF Method

Let us construct a partition $0 = t_0 < t_1 < t_2 \ldots < t_m = T$ of the time interval $[0, T]$ and define the time step $\tau_n = t_n - t_{n-1}$. We use the approximations $\boldsymbol{w}_h(t_n) \approx \boldsymbol{w}_h^n \in \boldsymbol{S}_{ht_n}^r$, $z(t_n) \approx z^n$, $n = 0, 1, \ldots$. Let us assume that $\boldsymbol{w}_h^n$, $n = 0, \ldots, m-1$, are already known. Then we introduce the functions $\hat{\boldsymbol{w}}_h^n = \boldsymbol{w}_h^n \circ \mathscr{A}_{t_n} \circ \mathscr{A}_{t_m}^{-1}$ for $n = m-1, m-2, \ldots$, which represents the transformation of $\boldsymbol{w}_h^n$ on the domain Ω_{ht_m}. The ALE derivative at time t_m is approximated by the backward finite difference formula (BDF) of order q:

$$\frac{D^{\mathscr{A}}\boldsymbol{w}_h}{Dt}(t_m) \approx \frac{D_{appr}^{\mathscr{A}}\boldsymbol{w}_h}{Dt}(t_m) = \alpha_0 \boldsymbol{w}_h^m + \sum_{l=1}^{q} \alpha_l \hat{\boldsymbol{w}}_h^{m-l}, \tag{11.37}$$

with coefficients α_l, $l = 0, \ldots, q$, depending on τ_{m-l}, $l = 0, \ldots, q-1$. In the beginning of the computation, when $m < q$, we approximate the ALE derivative by formulas of the lower order $q := m$. In nonlinear terms we use the extrapolation for the computation of the state $\overline{\boldsymbol{w}}_h^m$:

$$\overline{\boldsymbol{w}}_h^m = \sum_{l=1}^{q} \beta_l \hat{\boldsymbol{w}}_h^{m-l}, \tag{11.38}$$

where β_l, $l = 1, \ldots, q$, depend on τ_{m-l}, $l = 0, \ldots, q-1$. If $m < q$, then we apply extrapolation of order m. Namely, for $q = 1$ we have

$$\frac{D_{appr}^{\mathscr{A}}\boldsymbol{w}_h}{Dt}(t_m) = \frac{\boldsymbol{w}_h^m - \hat{\boldsymbol{w}}_h^{m-1}}{\tau_m} \tag{11.39}$$

and

$$\overline{\boldsymbol{w}}_h^m = \hat{\boldsymbol{w}}_h^{m-1}. \tag{11.40}$$

If $q = 2$, then

$$\begin{aligned} &\frac{D_{appr}^{\mathscr{A}}\boldsymbol{w}_h}{Dt}(t_m) \\ &= \frac{2\tau_m + \tau_{m-1}}{\tau_m(\tau_m + \tau_{m-1})}\boldsymbol{w}_h^{m+1} - \frac{\tau_m + \tau_{m-1}}{\tau_m \tau_{m-1}}\hat{\boldsymbol{w}}_h^m + \frac{\tau_m}{\tau_{m-1}(\tau_m + \tau_{m-1})}\hat{\boldsymbol{w}}_h^{m-1} \end{aligned} \tag{11.41}$$

and

$$\overline{\boldsymbol{w}}_h^m = \frac{\tau_m + \tau_{m-1}}{\tau_{m-1}} \hat{\boldsymbol{w}}_h^m - \frac{\tau_m}{\tau_{m-1}} \hat{\boldsymbol{w}}_h^{m-1}. \tag{11.42}$$

By the symbol $(\cdot,\cdot)_t$ we shall denote the scalar product in $L^2(\Omega_{ht})$, i.e.,

$$(\boldsymbol{w}_h, \boldsymbol{\varphi}_h)_t = \int_{\Omega_{ht}} \boldsymbol{w}_h \cdot \boldsymbol{\varphi}_h \, dx. \tag{11.43}$$

The resulting *BDF-DG scheme* has the following form: For each $m = 1, 2, \ldots$ we seek $\boldsymbol{w}_h^m \in \boldsymbol{S}_{ht_m}^r$ such that $\forall \boldsymbol{\varphi}_h \in \boldsymbol{S}_{ht_m}^r$,

$$\left(\frac{D_{appr}^{\mathscr{A}} \boldsymbol{w}_h}{Dt}(t_m), \boldsymbol{\varphi}_h \right)_{t_m} + \hat{b}_h(\overline{\boldsymbol{w}}_h^m, \boldsymbol{w}_h^m, \boldsymbol{\varphi}_h, t_m) + \hat{a}_h(\overline{\boldsymbol{w}}_h^m, \boldsymbol{w}_h^m, \boldsymbol{\varphi}_h, t_m) \tag{11.44}$$

$$+ J_h(\boldsymbol{w}_h^m, \boldsymbol{\varphi}_h, t_m) + d_h(\boldsymbol{w}_h^m, \boldsymbol{\varphi}_h, t_m) + \hat{\beta}_h(\overline{\boldsymbol{w}}_h^m, \boldsymbol{w}_h^m, \boldsymbol{\varphi}_h, t_m)$$

$$+ \hat{J}_h(\overline{\boldsymbol{w}}_h^m, \boldsymbol{w}_h^m, \boldsymbol{\varphi}_h, t_m) = \ell(\overline{\boldsymbol{w}}_B^m, \boldsymbol{\varphi}_h, t_m).$$

11.3.2.2 Space-Time Discontinuous Galerkin Method for the Flow Problem

We again consider a partition $0 = t_0 < t_1 < \ldots < t_M = T$ of the time interval $[0, T]$ and denote $I_m = (t_{m-1}, t_m)$, $\overline{I}_m = [t_{m-1}, t_m]$, $\tau_m = t_m - t_{m-1}$, for $m = 1, \ldots, M$. We define the space $\mathbf{S}_{h\tau}^{rq} = (S_{h\tau}^{rq})^4$, where

$$S_{h\tau}^{rq} = \left\{ \phi \,;\, \phi|_{I_m} = \sum_{i=0}^{q} \zeta_i \phi_i, \text{ where } \phi_i(t) \in S_{ht}^r, \text{ for } t \in I_m, \ \zeta_i \in P^q(I_m) \right\}$$

with integers $r, q \geq 1$. $P^q(I_m)$ denotes the space of all polynomials in t on I_m of degree $\leq q$ and the space S_{ht}^r is defined in (11.23). For $\boldsymbol{\varphi} \in \mathbf{S}_{h\tau}^{rq}$ we introduce the following notation:

$$\boldsymbol{\varphi}_m^{\pm} = \boldsymbol{\varphi}(t_m^{\pm}) = \lim_{t \to t_m \pm} \boldsymbol{\varphi}(t), \quad \{\boldsymbol{\varphi}\}_m = \boldsymbol{\varphi}_m^+ - \boldsymbol{\varphi}_m^-. \tag{11.45}$$

The derivation of the discrete problem can be carried out similarly as above. The difference is now that time t is considered continuous, test functions $\boldsymbol{\varphi}_{h\tau} \in \mathbf{S}_{h\tau}^{rq}$ are used and also the integration over I_m is applied. In order to bind the solution on intervals I_{m-1} and I_m, we augment the resulting identity by the penalty expression $\left(\{\boldsymbol{w}_{h\tau}\}_{m-1}, \boldsymbol{\varphi}_{h\tau}(t_{m-1}^+)\right)_{t_{m-1}}$. The initial state $\boldsymbol{w}_{h\tau}(0-) \in S_{h0}^p$ is defined as the $L^2(\Omega_{h0})$-projection of $\boldsymbol{w}^0$ on S_{h0}^p, i.e.,

$$(\boldsymbol{w}_{h\tau}(0-), \boldsymbol{\varphi}_h)_{t_0} = \left(\boldsymbol{w}^0, \boldsymbol{\varphi}_h\right)_{t_0} \quad \forall \boldsymbol{\varphi}_h \in \mathbf{S}_{h0}^p. \tag{11.46}$$

Moreover, we introduce the prolongation $\overline{\boldsymbol{w}}_{h\tau}(t)$ of $\boldsymbol{w}_{h\tau}|_{I_{m-1}}$ on the interval I_m. The approximate solution on the initial time interval I_1 is defined as a constant state $\boldsymbol{w}_{h\tau}(0-)$ given by (11.46). (The space-time DG technique with prolongation was analyzed theoretically in [21] on a scalar model problem.)

Now the *space-time DG (ST-DG) approximate solution* is defined as a function $\boldsymbol{w}_{h\tau} \in \boldsymbol{S}_{h\tau}^{rq}$ satisfying (11.46) and the following relation for $m = 1, \ldots, M$ and for all $\boldsymbol{\varphi}_h \in \boldsymbol{S}_{ht_m}^r$:

$$\begin{aligned}
\int_{I_m} \left(\left(\frac{D^{\mathscr{A}} \boldsymbol{w}_{h\tau}}{Dt}(t), \boldsymbol{\varphi}_{h\tau} \right)_t + \hat{a}_h(\overline{\boldsymbol{w}}_{h\tau}, \boldsymbol{w}_{h\tau}, \boldsymbol{\varphi}_{h\tau}, t) \right) \mathrm{d}t \\
+ \int_{I_m} \left(\hat{b}_h(\overline{\boldsymbol{w}}_{h\tau}, \boldsymbol{w}_{h\tau}, \boldsymbol{\varphi}_{h\tau}, t) + \int_{I_m} J_h(\boldsymbol{w}_{h\tau}, \boldsymbol{\varphi}_{h\tau}, t) \right) \mathrm{d}t \\
+ \int_{I_m} \left(\hat{\beta}_h(\overline{\boldsymbol{w}}_{h\tau}, \boldsymbol{w}_{h\tau}, \boldsymbol{\varphi}_{h\tau}, t) + \hat{J}_h(\overline{\boldsymbol{w}}_{h\tau}, \boldsymbol{w}_{h\tau}, \boldsymbol{\varphi}_{h\tau}, t) \right) \mathrm{d}t \\
+ (\{\boldsymbol{w}_{h\tau}\}_{m-1}, \boldsymbol{\varphi}_{h\tau}(t_{m-1}+)) \\
= \int_{I_m} \ell_h(\boldsymbol{w}_B, \boldsymbol{\varphi}_{h\tau}, t) \, \mathrm{d}t.
\end{aligned} \tag{11.47}$$

In practical computations, integrals appearing in the definitions of the forms $\hat{a}_h$, $\hat{b}_h$, d_h, J_h, $\hat{J}_h$ and $\hat{\beta}_h$ and also the time integrals are evaluated with the aid of quadrature formulas.

The linear algebraic systems equivalent to (11.44) and (11.47) are solved either by the direct solver UMFPACK [5] or by the GMRES method with block diagonal preconditioning.

11.3.3 Discretization of the Structural Problem

In [15], the elasticity problem was discretized in space by the conforming finite element method and for the time discretization the Newmark method was used. Here the technique for the numerical solution of the dynamical elasticity problem based on the DGM will be applied. Problem (11.10)–(11.14) can be reformulated into a couple of equations of the first order in time: Find functions $\boldsymbol{u}$ and $\boldsymbol{z} : \Omega^b \times [0, T] \to \mathbb{R}^2$ such that

$$\rho^b \frac{\partial \boldsymbol{z}}{\partial t} + c_M \rho^b \boldsymbol{z} - \operatorname{div} \boldsymbol{\sigma}(\boldsymbol{u}) - c_K \operatorname{div} \boldsymbol{\sigma}(\boldsymbol{z}) = \boldsymbol{f} \qquad \text{in } \Omega^b \times (0, T), \tag{11.48}$$

$$\frac{\partial \boldsymbol{u}}{\partial t} - \boldsymbol{z} = 0 \qquad \text{in } \Omega^b \times (0,T), \tag{11.49}$$

$$\boldsymbol{u} = \boldsymbol{u}_D \qquad \text{in } \Gamma_D^b \times (0,T), \tag{11.50}$$

$$\boldsymbol{\sigma}(\boldsymbol{u}) \cdot \boldsymbol{n} = \boldsymbol{g}_N \qquad \text{in } \Gamma_N^b \times (0,T), \tag{11.51}$$

$$\boldsymbol{u}(x,0) = \boldsymbol{u}_0(x), \qquad x \in \Omega^b, \tag{11.52}$$

$$\boldsymbol{z}(x,0) = \boldsymbol{z}_0(x), \qquad x \in \Omega^b. \tag{11.53}$$

Now we shall proceed in a similar way as in Sect. 11.3.1. By Ω_h^b we denote a polygonal approximation of the domain Ω^b. The symbols Γ_{Dh}^b, $\Gamma_{Nh}^b \subset \partial\Omega_h^b$ will approximate Γ_D^b and Γ_N^b. Let $\mathcal{T}_h^b$ be a partition of the closure $\overline{\Omega}_h^b$ of the domain Ω_h^b into a finite number of closed triangles K with mutually disjoint interiors. We assume that $\Gamma_{Dh}^b \cap \Gamma_{Nh}^b$ is formed by vertices of some $K \in \mathcal{T}_h^b$. We define the finite dimensional space

$$\boldsymbol{S}_{hs}^b = \left\{ v \in L^2(\Omega_h^b);\, v|_K \in P_s(K),\, K \in \mathcal{T}_h^b \right\}^2, \tag{11.54}$$

where $s > 0$ is an integer. Further, we use the partition $0 = t_0 < \cdots < t_M = T$ of the time interval $[0,T]$ with the steps $\tau_m = t_m - t_{m-1}$, $m = 1, \ldots, M$, introduced in Sect. 11.3. We shall use a similar notation as in Sect. 11.3.1. By $\mathcal{F}_h^b$ we denote the system of all faces of all elements $K \in \mathcal{T}_h^b$. Further, we define the set of all boundary, "Dirichlet", "Neumann" and inner faces by $\mathcal{F}_h^{bB} = \left\{\Gamma \in \mathcal{F}_h^b;\, \Gamma \subset \partial\Omega_h^b\right\}$, $\mathcal{F}_h^{bD} = \left\{\Gamma \in \mathcal{F}_h^b;\, \Gamma \subset \partial\Gamma_{Dh}^b\right\}$, $\mathcal{F}_h^{bN} = \left\{\Gamma \in \mathcal{F}_h^b;\, \Gamma \subset \partial\Gamma_{Nh}^b\right\}$, $\mathcal{F}_h^{bI} = \mathcal{F}_h^b \setminus \mathcal{F}_h^{bB}$, respectively. For each $\Gamma \in \mathcal{F}_h^b$ we define a unit normal vector $\boldsymbol{n}_\Gamma$. We assume that for $\Gamma \in \mathcal{F}_h^{bB}$ the normal $\boldsymbol{n}_\Gamma$ has the same orientation as the outer normal to $\partial\Omega_h^b$. We use the notation $\boldsymbol{\varphi}_\Gamma^{(L)}$, $\boldsymbol{\varphi}_\Gamma^{(R)}$, $\langle \boldsymbol{\varphi} \rangle_\Gamma$ and $[\boldsymbol{\varphi}]_\Gamma$ defined in the same way as in Sect. 11.3.1.

The discrete problem is obtained in a standard way. We multiply equations (11.48) and (11.49) by test functions $\boldsymbol{\varphi} \in \boldsymbol{S}_{hs}^b$, integrate over $K \in \mathcal{T}_h^b$, sum the resulting equations over all $K \in \mathcal{T}_h^b$ and add some expressions mutually vanishing. We get the forms

$$\begin{aligned} a_h^b(\boldsymbol{u},\boldsymbol{\varphi}) = & \sum_{K \in \mathcal{T}_h^b} \int_K \boldsymbol{\sigma}(\boldsymbol{u}) : \boldsymbol{e}(\boldsymbol{\varphi})\, \mathrm{d}x - \sum_{\Gamma \in \mathcal{F}_h^{bI}} \int_\Gamma (\langle \boldsymbol{\sigma}(\boldsymbol{u}) \rangle \cdot \boldsymbol{n}) \cdot [\boldsymbol{\varphi}]\, \mathrm{d}S \\ & - \sum_{\Gamma \in \mathcal{F}_h^{bD}} \int_\Gamma (\boldsymbol{\sigma}(\boldsymbol{u}) \cdot \boldsymbol{n}) \cdot \boldsymbol{\varphi}\, \mathrm{d}S - \theta \sum_{\Gamma \in \mathcal{F}_h^{bI}} \int_\Gamma (\langle \boldsymbol{\sigma}(\boldsymbol{\varphi}) \rangle \cdot \boldsymbol{n}) \cdot [\boldsymbol{u}]\, \mathrm{d}S, \\ & - \theta \sum_{\Gamma \in \mathcal{F}_h^{bD}} \int_\Gamma (\boldsymbol{\sigma}(\boldsymbol{\varphi}) \cdot \boldsymbol{n}) \cdot \boldsymbol{u}\, \mathrm{d}S, \end{aligned} \tag{11.55}$$

$$J_h^b(\boldsymbol{u},\boldsymbol{\varphi}) = \sum_{\Gamma\in\mathscr{F}_h^{bI}} \int_\Gamma \frac{C_W^b}{h_\Gamma}[\boldsymbol{u}]\cdot[\boldsymbol{\varphi}]\,\mathrm{d}S + \sum_{\Gamma\in\mathscr{F}_h^{bD}} \int_\Gamma \frac{C_W^b}{h_\Gamma}\boldsymbol{u}\cdot\boldsymbol{\varphi}\,\mathrm{d}S, \tag{11.56}$$

$$\ell_h^b(\boldsymbol{\varphi})(t) = \sum_{K\in\mathscr{T}_h^b} \int_K \boldsymbol{f}(t)\cdot\boldsymbol{\varphi}\,\mathrm{d}x + \sum_{\Gamma\in\mathscr{F}_h^{bN}} \int_\Gamma \boldsymbol{g}_N(t)\cdot\boldsymbol{\varphi}\,\mathrm{d}S \tag{11.57}$$

$$- \theta \sum_{\Gamma\in\mathscr{F}_h^{bD}} \int_\Gamma (\boldsymbol{\sigma}(\boldsymbol{\varphi})\cdot\boldsymbol{n})\cdot\boldsymbol{u}_D(t)\,\mathrm{d}S + \sum_{\Gamma\in\mathscr{F}_h^{bD}} \int_\Gamma \frac{C_W^b}{h_\Gamma}\boldsymbol{u}_D(t)\cdot\boldsymbol{\varphi}\,\mathrm{d}S,$$

$$\ell_h^d(\boldsymbol{\varphi})(t) = \sum_{\Gamma\in\mathscr{F}_h^{bN}} \int_\Gamma \frac{\partial \boldsymbol{g}_N(t)}{\partial t}\cdot\boldsymbol{\varphi}\,\mathrm{d}S \tag{11.58}$$

$$- \theta \sum_{\Gamma\in\mathscr{F}_h^{bD}} \int_\Gamma (\boldsymbol{\sigma}(\boldsymbol{\varphi})\cdot\boldsymbol{n})\cdot\frac{\partial \boldsymbol{u}_D(t)}{\partial t}\,\mathrm{d}S + \sum_{\Gamma\in\mathscr{F}_h^{bD}} \int_\Gamma \frac{C_W^b}{h_\Gamma}\frac{\partial \boldsymbol{u}_D(t)}{\partial t}\cdot\boldsymbol{\varphi}\,\mathrm{d}S,$$

where $\theta = 1, 0$ or -1 and C_W^b is a positive sufficiently large constant. Moreover, we use the notation

$$(\boldsymbol{u},\boldsymbol{\varphi})_{\Omega_h^b} = \int_{\Omega_h^b} \boldsymbol{u}\cdot\boldsymbol{\varphi}\,\mathrm{d}x. \tag{11.59}$$

Now it is possible to introduce discrete problems. We shall consider several schemes according to the type of the time discretization.

11.3.3.1 Simple Backward Difference Scheme for the Time Discretization of the Structural Problem

Let the time partition be uniform, i.e. $\tau_m = \tau$ for $m = 1,\dots,M$. We use the approximations $\boldsymbol{u}(t_m) \approx \boldsymbol{u}_h^m \in \boldsymbol{S}_{hs}^b$,

$$\frac{\partial \boldsymbol{u}}{\partial t}(t_m) \approx \frac{\boldsymbol{u}_h^m - \boldsymbol{u}_h^{m-1}}{\tau}, \tag{11.60}$$

$$\frac{\partial^2 \boldsymbol{u}}{\partial t^2}(t_m) \approx \frac{\boldsymbol{u}_h^m - 2\boldsymbol{u}_h^{m-1} + \boldsymbol{u}_h^{m-2}}{\tau^2}. \tag{11.61}$$

Then the approximate solution is obtained from the scheme based on problem (11.10)–(11.14). For $m = 2,\dots,M$ find $\boldsymbol{u}_h^m \in \boldsymbol{S}_{hs}^b$ such that

$$\text{(a)}\quad \left(\rho(t_m)\frac{\boldsymbol{u}_h^m - 2\boldsymbol{u}_h^{m-1} + \boldsymbol{u}_h^{m-2}}{\tau^2},\ \boldsymbol{\varphi}_h\right)_{\Omega_h^b} + c_M\left(\rho^b(t_m)\frac{\boldsymbol{u}_h^m - \boldsymbol{u}_h^{m-1}}{\tau},\ \boldsymbol{\varphi}_h\right)_{\Omega_h^b} \tag{11.62}$$

$$+a_h^b\left(\boldsymbol{u}_h^m,\boldsymbol{v}_h\right)+c_K a_h^b\left(\frac{\boldsymbol{u}_h^m-\boldsymbol{u}_h^{m-1}}{\tau},\boldsymbol{v}_h\right)+J_h^b\left(\boldsymbol{u}_h^m,\boldsymbol{\varphi}_h\right)$$

$$=\ell_h^b(\boldsymbol{\varphi}_h)\,(t_m)+c_K\ell_h^d(\boldsymbol{\varphi}_h)\,(t_m)\quad\forall\,\boldsymbol{\varphi}_h\in\boldsymbol{S}_{hs}^b,$$

(b) $\left(\boldsymbol{u}_h^0,\boldsymbol{\varphi}_h\right)_{\Omega_h^b}=\left(\boldsymbol{u}^0,\boldsymbol{\varphi}_h\right)_{\Omega_h^b}\quad\forall\,\boldsymbol{\varphi}_h\in\boldsymbol{S}_{hs}^b,$

(c) $\left(\boldsymbol{u}_h^1,\boldsymbol{\varphi}_h\right)_{\Omega_h^b}=\left(\boldsymbol{z}_0+\tau\boldsymbol{u}_h^0,\boldsymbol{\varphi}_h\right)_{\Omega_h^b}\quad\forall\,\boldsymbol{\varphi}_h\in\boldsymbol{S}_{hs}^b.$

11.3.3.2 BDF Method Based on Splitting of the Structural Problem

In this case we use formulation (11.48)–(11.53) and use the approximation analogous to (11.37):

$$\frac{\partial\boldsymbol{u}}{\partial t}(t_m)\approx\frac{D_{\text{appr}}\boldsymbol{u}_h^m}{Dt}=\alpha_0\,\boldsymbol{u}_h^m+\sum_{\ell=1}^{q}\alpha_\ell\boldsymbol{u}_h^{m-\ell}.\tag{11.63}$$

The coefficients α_ℓ depend on time steps $\tau_m,\tau_{m-1},\ldots$. See, e.g., (11.39) and (11.41). Similar formula is used for the approximation of $\partial z/\partial t$. Then we get the discrete problem to find $\boldsymbol{u}_h^m,\boldsymbol{z}_h^m\in\boldsymbol{S}_{hs}^b$ such that for $m=1,\ldots,M$,

(a) $\left(\frac{D_{\text{appr}}\boldsymbol{z}_h^m}{Dt},\boldsymbol{\varphi}_h\right)_{\Omega_h^b}+c_M\left(\rho^b(t_m)\,\boldsymbol{z}_h^m,\boldsymbol{\varphi}_h\right)_{\Omega_h^b}+a_h^b\left(\boldsymbol{u}^m,\boldsymbol{\varphi}_h\right)+c_K a_h^b\left(\boldsymbol{z}_h^m,\boldsymbol{\varphi}_h\right)$
$+J_h^b\left(\boldsymbol{u}^m,\boldsymbol{\varphi}_h\right)=\ell_h^b(\boldsymbol{\varphi}_h)\,(t_m)+c_K\ell_h^d(\boldsymbol{\varphi}_h)\,(t_m)\quad\forall\,\boldsymbol{\varphi}_h\in\boldsymbol{S}_{hs}^b,$

(b) $\left(\frac{D_{\text{appr}}\boldsymbol{u}_h^m}{Dt},\boldsymbol{\varphi}_h\right)_{\Omega_h^b}-\left(\boldsymbol{z}_h^m,\boldsymbol{\varphi}_h\right)_{\Omega_h^b}=0\quad\forall\,\boldsymbol{\varphi}_h\in\boldsymbol{S}_{hs}^b,$

(c) $\left(\boldsymbol{u}_h^0,\boldsymbol{\varphi}_h\right)_{\Omega_h^b}=\left(\boldsymbol{u}^0,\boldsymbol{\varphi}_h\right)_{\Omega_h^b}\quad\forall\,\boldsymbol{\varphi}_h\in\boldsymbol{S}_{hs}^b,$

(d) $\left(\boldsymbol{z}_h^0,\boldsymbol{\varphi}_h\right)_{\Omega_h^b}=(z_0,\boldsymbol{\varphi}_h)_{\Omega_h^b}\quad\forall\,\boldsymbol{\varphi}_h\in\boldsymbol{S}_{hs}^b.$

11.3.3.3 Space-Time DG Method for the Structural Problem

We use a similar notation as in Sect. 11.3.2.2. An approximate solution of problem (11.48)–(11.49), i.e., the approximations of the functions $\boldsymbol{u}$, $\boldsymbol{z}$ will be sought in the space of piecewise polynomial vector functions $\boldsymbol{S}_{h\tau}^{b,sq}=[S_{h\tau}^{b,sq}]^2$, where

$$S_{h\tau}^{b,sq}=\left\{v\in L^2(\Omega_h^b)\times(0,T);\,v|_{I_m}=\sum_{i=0}^{q}t^i\varphi_i\ \text{with}\ \varphi_i\in S_{hs}^b,m=1,\ldots,M\right\}.\tag{11.64}$$

We introduce the one-sided limits and jump of a function $\boldsymbol{\varphi}\in[S_{h\tau}^{b,sq}]^2$ at time t_m by (11.45).

Now, the approximate space-time DG solution of problem (11.48)–(11.52) is defined as a couple $\boldsymbol{u}_{h\tau}, z_{h\tau} \in \boldsymbol{S}_{h\tau}^{b,sq}$ such that

$$
\begin{aligned}
\text{(a)} \int_{I_m} \Big(\rho^b \big(\frac{\partial \boldsymbol{z}_{h\tau}}{\partial t}, \boldsymbol{\varphi}_{h\tau}\big)_{\Omega_h^b} + c_M \left(\rho^b \boldsymbol{z}_{h\tau}, \boldsymbol{\varphi}_{h\tau}\right)_{\Omega_h^b} \Big) + a_h^b(\boldsymbol{u}_{h\tau}, \boldsymbol{\varphi}_{h\tau}) + c_K a_h^b(z_{h\tau}, \boldsymbol{\varphi}_{h\tau}) \\
+ J_h^b(\boldsymbol{u}_{h\tau}, \boldsymbol{\varphi}_{h\tau}) \Big) \,\mathrm{d}t + (\{\boldsymbol{u}_{h\tau}\}_{m-1}, \boldsymbol{\varphi}_{h\tau}(t_{m-1}+))_{\Omega_h^b} \\
= \int_{I_m} \ell(\boldsymbol{\varphi}_{h\tau}) \,\mathrm{d}t + c_K \int_{I_m} \ell_h^d(\boldsymbol{\varphi}_{h\tau}) \,\mathrm{d}t \;\; \forall \boldsymbol{\varphi}_{h\tau} \in \boldsymbol{S}_{h\tau}^{b,sq}, \\
\text{(b)} \int_{I_m} \Big(\Big(\frac{\partial \boldsymbol{u}_{h\tau}}{\partial t}, \boldsymbol{\varphi}_{h\tau} \Big)_{\Omega_h^b} - (z_{h\tau}, \boldsymbol{\varphi}_{h\tau})_{\Omega_h^b} \Big) \,\mathrm{d}t \\
+ (\{\boldsymbol{u}_{h\tau}\}_{m-1}, \boldsymbol{\varphi}_{h\tau}(t_{m-1}+))_{\Omega_h^b} = 0 \quad \forall \boldsymbol{\varphi}_{h\tau} \in \boldsymbol{S}_{h\tau}^{b,sq}, \\
m = 1, \ldots, M.
\end{aligned}
$$

Similarly as in (11.46) we define the initial states $\boldsymbol{u}_h(0-), z_h(0-) \in \boldsymbol{S}_{hs}^b$ by

$$
\begin{aligned}
(\boldsymbol{u}_h(0-), \boldsymbol{\varphi}_h)_{\Omega_h^b} &= (\boldsymbol{u}^0, \boldsymbol{\varphi}_h)_{\Omega_h^b} \quad \forall \boldsymbol{\varphi}_h \in \boldsymbol{S}_{hs}^b, \\
(z_h(0-), \boldsymbol{\varphi}_h)_{\Omega_h^b} &= (z^0, \boldsymbol{\varphi}_h)_{\Omega_h^b} \quad \forall \boldsymbol{\varphi}_h \in \boldsymbol{S}_{hs}^b.
\end{aligned} \tag{11.65}
$$

11.3.3.4 Numerical Test for the DG Methods Applied to the Dynamical Elasticity Problem

We assume an elastic material, which at rest occupies a 2 cm long and 2 mm thick rectangular-shape domain. The left edge of the material is fixed, while the right edge is exposed for a short period to a pressure force, which pulls the material in the direction of the x_1-axis. The problem is defined by the following data:

$$
\begin{aligned}
&T = 0.05\,\mathrm{s}, \quad \Omega^b = (-0.01, 0.01) \times (-0.001, 0.001)\,[\mathrm{m}], \\
&\Gamma_D^b = \left\{x = (x_1, x_2) \in \partial\Omega^b; x_1 = -0.001\right\}, \quad \Gamma_N^b = \partial\Omega^b \backslash \Gamma_D^b, \\
&\boldsymbol{f} = (0,0)^T \text{ in } \Omega^b \times (0,T), \; \boldsymbol{u}_D = (0,0)^T \text{ in } \Gamma_D^b \times (0,T), \\
&\boldsymbol{g}_N = \begin{cases} (2500,0)^T & \text{in } \Gamma_N^b \times [0, 0.02)\,[\mathrm{Pa}], \\ (0,0) & \text{in } \Gamma_N^b \times [0.02, T], \end{cases} \\
&\boldsymbol{u}^0 = (0,0)^T \quad \text{in } \Omega^b, \quad \boldsymbol{z}^0 = (0,0)^T \quad \text{in } \Omega^b.
\end{aligned}
$$

We assume that the material constants are given by $\rho^b = 1100\,\mathrm{kg.m}^{-3}, c_M = 0.1\,\mathrm{s}^{-1}, c_K = 0\,\mathrm{s}^{-1}$, $E^b = 10^5\,\mathrm{Pa}$ and $\sigma^b = 0.4$.

In the numerical experiment we compared the approximate solutions obtained by the space-time DG method with the space DG discretization combined with the second-order BDF time discretization. In both cases $\theta = -1$. We have chosen

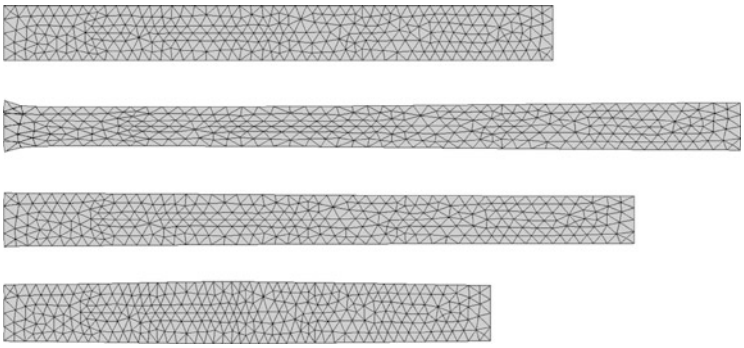


Fig. 11.2 Numerical approximation of the displacement function $\boldsymbol{u}$ for time instants $t = 0$, $t = 0.01$, $t = 0.02$ and $t = 0.03$ s (from the top to the bottom)

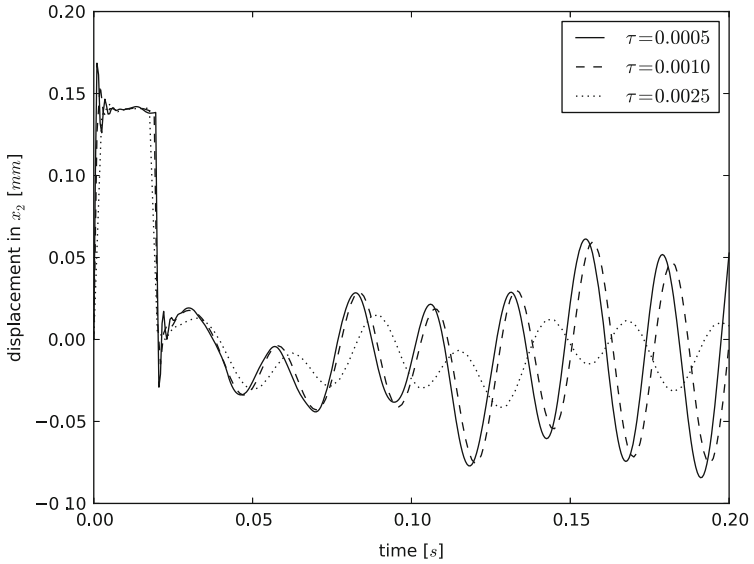


Fig. 11.3 Evolution of the x_2-component of the displacement function $\boldsymbol{u}$ at $x_1 = 0.001$, $x_2 = 0.01$ in time (method: BDF2)

three different time steps, $\tau = 0.0005$, 0.0010 and 0.0025. For both schemes we worked with the linear approximation in space, i.e., we set $s = 1$ (nonsymmetric version) and $C_W^b = 10^5$. The triangulation $\mathcal{T}_h^b$ of the domain Ω^b consists of 576 elements. Figure 11.2 shows the numerical approximation of the displacement function $\boldsymbol{u}$ at several time instants, while Figs. 11.3 and 11.4 show the comparison of the time evolution of the component u_2 of the displacement function $\boldsymbol{u}$ at a fixed point (x_1, x_2) using two different time schemes: backward-difference formula of the second order (BDF2) and the space-time DG method with the time polynomial

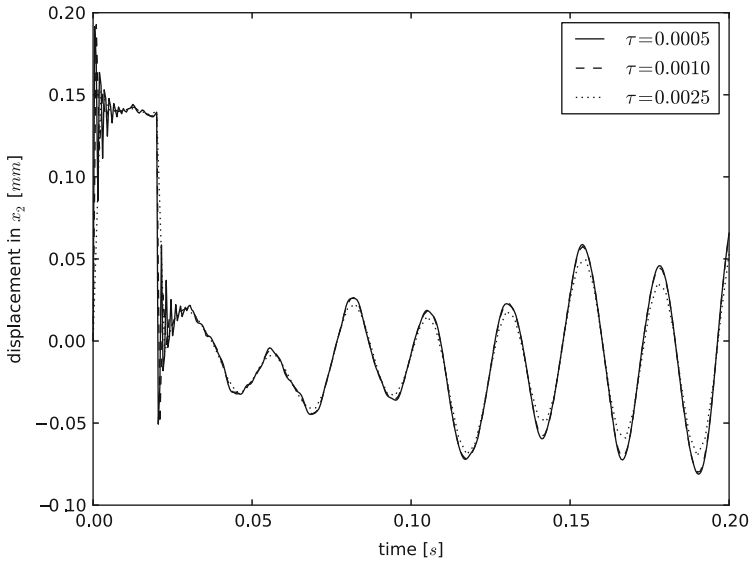


Fig. 11.4 Time evolution of the component u_2 of the displacement function $\boldsymbol{u}$ at $(x_1, x_2) = (0.001, 0.01)$ (method: ST1)

degree $q = 1$ (ST1). From this figure we can conclude that the space-time DG method converges faster with the decreasing time step than the BDF method.

The numerical solution of the problem (11.20)–(11.21) is carried out by the conforming piecewise linear finite elements.

11.3.4 Coupling Procedure

In the solution of the complete coupled fluid-structure interaction problem it is necessary to apply a suitable coupling procedure. In our case we apply two different types of algorithms. First, the so-called strong coupling will be presented.

1. Assume that the approximate solution of the flow problem and the deformation of the structure $\boldsymbol{u}_k$ on the time level t_k are known.
2. Set $\boldsymbol{u}_{k+1}^0 := \boldsymbol{u}_k$, $l := 1$ and apply the iterative process:
   (a) Compute the stress tensor τ_{ij}^f and the aerodynamical force acting on the structure and transform it to the interface Γ_N^b.
   (b) Solve the elasticity problem, compute the deformation $\boldsymbol{u}_{k+1}^l$ and the approximation $\Omega_{t_{k+1}}^l$ of the domain occupied by the fluid at time t_{k+1}.
   (c) Determine the ALE mapping $\mathscr{A}_{t_{k+1}}^l$ and approximate the domain velocity $\boldsymbol{z}_{k+1}^l$.

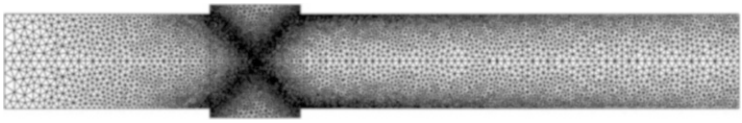


Fig. 11.5 Computational mesh

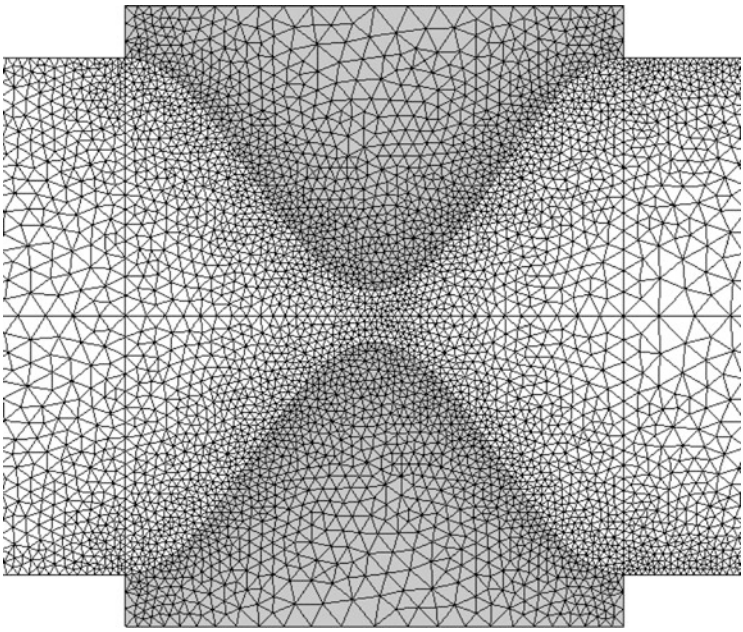


Fig. 11.6 Detail of the computational mesh

(d) Solve the flow problem on the approximation $\Omega^l_{t_{k+1}}$.
(e) If the variation $\boldsymbol{u}^l_{k+1} - \boldsymbol{u}^{l-1}_{k+1}$ of the displacement is larger than the prescribed tolerance, go to (a) and $l := l + 1$. Else $k := k + 1$ and go to (2).

In order to obtain the second type of the algorithm, the weak (loose) coupling, in step (e) we set $k := k + 1$ and go to (2) already in the case when $l = 1$.

11.3.5 Numerical Results

In order to show the applicability and robustness of the developed FSI technique, we present here results of numerical experiments.

First, we consider the model of flow through a channel with two bumps which represent time dependent boundaries between the flow and a simplified model of vocal folds (see Figs. 11.5 and 11.6). The numerical experiments were carried

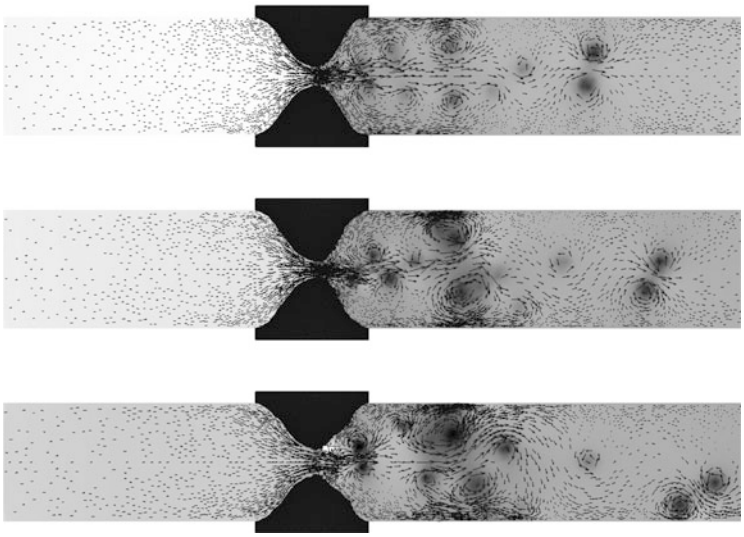


Fig. 11.7 The visualization of the deformation of the elastic bodies, velocity vectors and pressure at different time instants

out for the following data: magnitude of the inlet velocity $v_{in} = 4$ m.s^{-1}, the viscosity $\mu = 15 \cdot 10^{-6}\,\mathrm{kg\,m^{-1}\,s^{-1}}$, the inlet density $\rho_{in} = 1.225\,\mathrm{kg\,m^{-3}}$, the outlet pressure $p_{out} = 97{,}611$ Pa, the Reynolds number $Re = \rho_{in} v_{in} H/\mu = 5227$, heat conduction coefficient $k = 2.428 \cdot 10^{-2}\,\mathrm{kg\,m\,s^{-2}\,K^{-1}}$, the specific heat $c_v = 721.428\,\mathrm{m^2\,s^{-2}\,K^{-1}}$, the Poisson adiabatic constant $\gamma = 1.4$. The inlet Mach number is $M_{in} = 0.012$. The parameter of the computational accuracy of the GMRES solver was 10^{-5}. The Young modulus and the Poisson ratio have values $E^b = 25{,}000$ Pa and $\sigma^b = 0.4$, respectively, the structural damping coefficients are chosen as $c_M = 100\,\mathrm{s^{-1}}$ and $c_K = 0$ s and the material density $\rho^b = 1{,}040\,\mathrm{kg\,m^{-3}}$. The used time step was 10^{-5} s.

Fluid flow is solved by the ST-DGM with quadratic polynomials in space and linear polynomials in time. For the elasticity problem we also used the ST-DGM, but with linear polynomials in space and constant polynomials in time. For both problems the non-symmetric version (NIPG) was used. For flow problem we set $C_W = 1{,}000$ on the interior elements and $C_W = 10{,}000$ on the boundary elements in order to keep the prescribed Dirichlet boundary conditions, particularly in the boundary layer. For elasticity we set $C_W^b = 10^5$ in order to match the magnitude of the Lamè parametres.

Figure 11.7 shows the visualization of the deformation of the elastic bodies, velocity vectors and pressure at different time instants.

In the second example we consider a 2D model of gas flow past an elastic airfoil. (This is the continuation of results from [9, 18].) For testing our method we assume that the material of the airfoil is very soft. It is characterized by the Lamè parametres

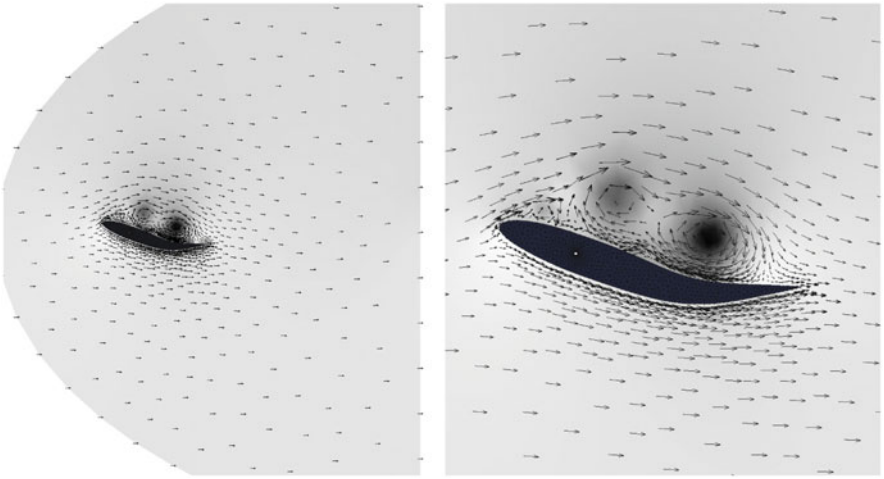


Fig. 11.8 The visualization of the flow past an elastic airfoil: velocity vectors and pressure

$\lambda^b = 2 \cdot 10^7$ Pa and $\mu^b = 5 \cdot 10^6$ Pa. The structural damping coefficients are chosen as $c_M = 0.1\,\mathrm{s}^{-1}$ and $c_K = 0.1$ s and the material density is given by $\rho^b = 10^4\,\mathrm{kg\,m}^{-3}$.

The fluid flow simulation was carried out using the following data: $\mu = 1.72 \cdot 10^{-5}\,\mathrm{kg\,m}^{-1}$.s, far-field pressure $p = 101{,}250$ Pa, far-field density $\rho = 1.225\,\mathrm{kg\,m}^{-3}$, Poisson adiabatic constant $\gamma = 1.4$, specific heat $c_v = 721.428\,\mathrm{m}^2\,\mathrm{s}^{-2}\,\mathrm{K}^{-1}$, heat conduction coefficient $k = 2.428 \cdot 10^{-2}\,\mathrm{kg\,m\,.s}^{-2}\,\mathrm{K}^{-1}$. The far-field velocity was $40\,\mathrm{m\,s}^{-1}$.

Fluid flow and the elasticity problem is again solved by the ST-DGM with the same settings as in the previous experiment, but for elasticity we set $C_W^b = 10^{10}$ in order to match the magnitude of the Lamè parametres and we use the time step $\tau = 2.25 \cdot 10^{-6}$ s. Figures 11.8 and 11.9 show the visualization of the deformed airfoil, the pressure and velocity fields.

11.4 Conclusion

The paper presents a description of a numerical method for the solution of compressible flow in time-dependent domains and the applications to the simulation of fluid-structure interaction. The gas flow is described by the 2D compressible Navier-Stokes equations in the ALE formulation allowing to take into account the time dependence of the computational domain. The flow problem is coupled with the structural problem represented by the system of dynamical linear elasticity. The space discretization is carried out by the discontinuous Galerkin method. For the time discretization the techniques based on the backward difference formula and

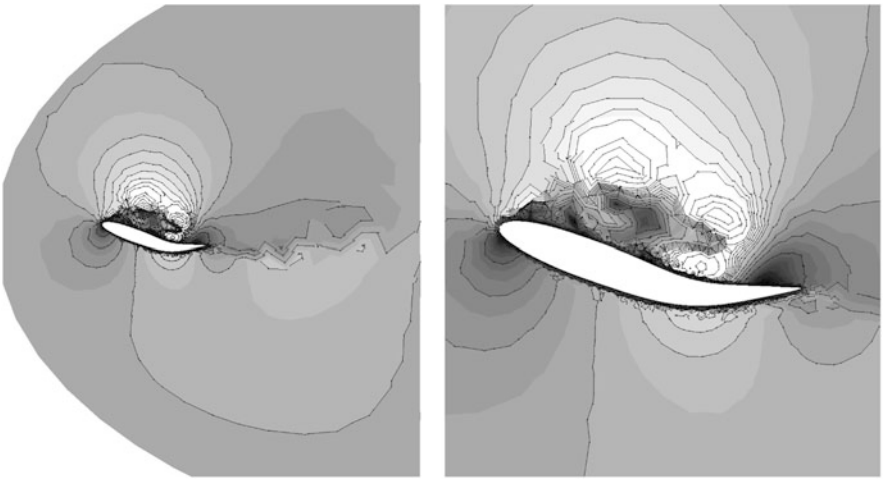


Fig. 11.9 The visualization of the flow past an elastic airfoil: velocity magnitude isolines

discontinuous Galerkin method are applied. This means that we get two methods: backward difference formula-discontinuous Galerkin method (BDF-DGM) and space-time discontinuous Galerkin method (ST-DGM).

Numerical experiments show that the method is robust with respect to a wide range of Mach numbers and Reynolds numbers. We tested here the convergence of the methods for the solution of the structural problem in dependence on a decreasing time step. The results show that the ST-DGM method is more robust than the BDF-DG scheme. Then the presented technique was applied to the problem of the interaction of compressible flow and an elastic structure.

There are the following subjects for further work:

- Realization of further tests of the developed techniques,
- Solution of problems with large deformations,
- Comparison of obtained results with wind tunnel experiments,
- Theoretical analysis of qualitative properties (e.g. stability, convergence) of the numerical methods.

Acknowledgements This work was supported by the grants P101/11/0207 (J. Horáček) and 13-00522S (M. Feistauer) of the Czech Science Foundation. The work of M. Hadrava and A. Kosík was supported by the grant SVV-2014-260106, financed by the Charles University in Prague.

References

1. Bassi, F., Rebay, S.: High-order accurate discontinuous finite element solution of the 2D Euler equations. J. Comput. Phys. **138**, 251–285 (1997)

2. Baumann, C.E., Oden, J.T.: A discontinuous *hp* finite element method for the Euler and Navier-Stokes equations. Int. J. Numer. Methods Fluids **31**, 79–95 (1999)
3. Bisplinghoff, R.L., Ashley, H., Halfman, R.L.: Aeroelasticity. Dover, New York (1996)
4. Česenek, J., Feistauer, M., Kosík, A.: DGFEM for the analysis of airfoil vibrations induced by compressible flow. ZAMM – Z. Angew. Math. Mech. **93**(6–7), 387–402 (2013)
5. Davis, T.A., Duff, I.S.: A combined unifrontal/multifrontal method for unsymmetric sparse matrices. ACM Trans. Math. Softw. **25**, 1–19 (1999)
6. Dolejší, V.: Semi-implicit interior penalty discontinuous Galerkin methods for viscous compressible flows. Commun. Comput. Phys. **4**, 231–274 (2008)
7. Dolejší, V., Feistauer, M., Schwab, C.: On some aspects of the discontinuous Galerkin finite element method for conservation laws. Math. Comput. Simul. **61**, 333–346 (2003)
8. Dowell, E.H.: A Modern Course in Aeroelasticity. Kluwer, Dodrecht (1995)
9. Dubcová, L., Feistauer, M., Horáček, J., Sváček, P.: Numerical simulation of interaction between turbulent flow and a vibrating airfoil. Comput. Vis. Sci. **12**, 207–225 (2009)
10. Feistauer, M.: Mathematical Methods in Fluid Dynamics. Longman, Harlow (1993)
11. Feistauer, M., Kučera, V.: On a robust discontinuous Galerkin technique for the solution of compressible flow. J. Comput. Phys. **224**, 208–221 (2007)
12. Feistauer, M., Felcman, J., Straškraba, I.: Mathematical and Computational Methods for Compressible Flow. Clarendon Press, Oxford (2003)
13. Feistauer, M., Česenek, J., Horáček, J., Kučera, V., Prokopová, J.: DGFEM for the numerical solution of compressible flow in time dependent domains and applications to fluid-structure interaction. In: Pereira, J.C.F., Sequeira, A. (eds.) Proceedings of the 5th European Conference on Computational Fluid Dynamics ECCOMAS CFD 2010, Lisbon (2010). ISBN 978-989-96778-1-4 (published ellectronically)
14. Feistauer, M., Horáček, J., Kučera, V., Prokopová, J.: On numerical solution of compressible flow in time-dependent domains. Math. Bohem. **137**, 1–16 (2012)
15. Feistauer, M., Hasnedlová-Prokopová, J., Horáček, J., Kosík, A., Kučera, V.: DGFEM for dynamical systems describing interaction of compressible fluid and structures. J. Comput. Appl. Math. **254**, 17–30 (2013)
16. Naudasher, E., Rockwell, D.: Flow-Induced Vibrations. A.A. Balkema, Rotterdam (1994)
17. Nomura, T., Hughes, T.J.R.: An arbitrary Lagrangian-Eulerian finite element method for interaction of fluid and a rigid body. Comput. Methods Appl. Mech. Eng. **95**, 115–138 (1992)
18. Sváček, P., Feistauer, M., Horáček, J.: Numerical simulation of flow induced airfoil vibrations with large amplitudes. J. Fluids Struct. **23**, 391–411 (2007)
19. van der Vegt, J.J.W., van der Ven, H.: Space-time discontinuous Galerkin finite element method with dynamic grid motion for inviscid compressible flow. J. Comput. Phys. **182**, 546–585 (2002)
20. Vijayasundaram, G.: Transonic flow simulation using upstream centered scheme of Godunov type in finite elements. J. Comput. Phys. **63**, 416–433 (1986)
21. Vlasák, M., Dolejší, V., Hájek, J.: A priori estimates of an extrapolated space-time discontinuous Galerkin method for nonlinear convection-diffusion problems. Numer. Methods Potential Differ. Eq. **27**, 1456–1482 (2011)

Editorial Policy

1. Volumes in the following three categories will be published in LNCSE:

i) Research monographs
ii) Tutorials
iii) Conference proceedings

Those considering a book which might be suitable for the series are strongly advised to contact the publisher or the series editors at an early stage.

2. Categories i) and ii). Tutorials are lecture notes typically arising via summer schools or similar events, which are used to teach graduate students. These categories will be emphasized by Lecture Notes in Computational Science and Engineering. **Submissions by interdisciplinary teams of authors are encouraged.** The goal is to report new developments – quickly, informally, and in a way that will make them accessible to non-specialists. In the evaluation of submissions timeliness of the work is an important criterion. Texts should be well-rounded, well-written and reasonably self-contained. In most cases the work will contain results of others as well as those of the author(s). In each case the author(s) should provide sufficient motivation, examples, and applications. In this respect, Ph.D. theses will usually be deemed unsuitable for the Lecture Notes series. Proposals for volumes in these categories should be submitted either to one of the series editors or to Springer-Verlag, Heidelberg, and will be refereed. A provisional judgement on the acceptability of a project can be based on partial information about the work: a detailed outline describing the contents of each chapter, the estimated length, a bibliography, and one or two sample chapters – or a first draft. A final decision whether to accept will rest on an evaluation of the completed work which should include

– at least 100 pages of text;
– a table of contents;
– an informative introduction perhaps with some historical remarks which should be accessible to readers unfamiliar with the topic treated;
– a subject index.

3. Category iii). Conference proceedings will be considered for publication provided that they are both of exceptional interest and devoted to a single topic. One (or more) expert participants will act as the scientific editor(s) of the volume. They select the papers which are suitable for inclusion and have them individually refereed as for a journal. Papers not closely related to the central topic are to be excluded. Organizers should contact the Editor for CSE at Springer at the planning stage, see *Addresses* below.

In exceptional cases some other multi-author-volumes may be considered in this category.

4. Only works in English will be considered. For evaluation purposes, manuscripts may be submitted in print or electronic form, in the latter case, preferably as pdf- or zipped ps-files. Authors are requested to use the LaTeX style files available from Springer at http://www.springer.com/authors/book+authors/helpdesk?SGWID=0-1723113-12-971304-0 (Click on Templates → LaTeX → monographs or contributed books).

For categories ii) and iii) we strongly recommend that all contributions in a volume be written in the same LaTeX version, preferably LaTeX2e. Electronic material can be included if appropriate. Please contact the publisher.

Careful preparation of the manuscripts will help keep production time short besides ensuring satisfactory appearance of the finished book in print and online.

5. The following terms and conditions hold. Categories i), ii) and iii):

Authors receive 50 free copies of their book. No royalty is paid.
Volume editors receive a total of 50 free copies of their volume to be shared with authors, but no royalties.

Authors and volume editors are entitled to a discount of 33.3 % on the price of Springer books purchased for their personal use, if ordering directly from Springer.

6. Springer secures the copyright for each volume.

Addresses:

Timothy J. Barth
NASA Ames Research Center
NAS Division
Moffett Field, CA 94035, USA
barth@nas.nasa.gov

Michael Griebel
Institut für Numerische Simulation
der Universität Bonn
Wegelerstr. 6
53115 Bonn, Germany
griebel@ins.uni-bonn.de

David E. Keyes
Mathematical and Computer Sciences
and Engineering
King Abdullah University of Science
and Technology
P.O. Box 55455
Jeddah 21534, Saudi Arabia
david.keyes@kaust.edu.sa

and

Department of Applied Physics
and Applied Mathematics
Columbia University
500 W. 120 th Street
New York, NY 10027, USA
kd2112@columbia.edu

Risto M. Nieminen
Department of Applied Physics
Aalto University School of Science
and Technology
00076 Aalto, Finland
risto.nieminen@aalto.fi

Dirk Roose
Department of Computer Science
Katholieke Universiteit Leuven
Celestijnenlaan 200A
3001 Leuven-Heverlee, Belgium
dirk.roose@cs.kuleuven.be

Tamar Schlick
Department of Chemistry
and Courant Institute
of Mathematical Sciences
New York University
251 Mercer Street
New York, NY 10012, USA
schlick@nyu.edu

Editor for Computational Science
and Engineering at Springer:
Martin Peters
Springer-Verlag
Mathematics Editorial IV
Tiergartenstrasse 17
69121 Heidelberg, Germany
martin.peters@springer.com

Lecture Notes in Computational Science and Engineering

1. D. Funaro, *Spectral Elements for Transport-Dominated Equations.*
2. H.P. Langtangen, *Computational Partial Differential Equations.* Numerical Methods and Diffpack Programming.
3. W. Hackbusch, G. Wittum (eds.), *Multigrid Methods V.*
4. P. Deuflhard, J. Hermans, B. Leimkuhler, A.E. Mark, S. Reich, R.D. Skeel (eds.), *Computational Molecular Dynamics: Challenges, Methods, Ideas.*
5. D. Kröner, M. Ohlberger, C. Rohde (eds.), *An Introduction to Recent Developments in Theory and Numerics for Conservation Laws.*
6. S. Turek, *Efficient Solvers for Incompressible Flow Problems.* An Algorithmic and Computational Approach.
7. R. von Schwerin, ***M**ulti **B**ody **S**ystem **SIM**ulation.* Numerical Methods, Algorithms, and Software.
8. H.-J. Bungartz, F. Durst, C. Zenger (eds.), *High Performance Scientific and Engineering Computing.*
9. T.J. Barth, H. Deconinck (eds.), *High-Order Methods for Computational Physics.*
10. H.P. Langtangen, A.M. Bruaset, E. Quak (eds.), *Advances in Software Tools for Scientific Computing.*
11. B. Cockburn, G.E. Karniadakis, C.-W. Shu (eds.), *Discontinuous Galerkin Methods.* Theory, Computation and Applications.
12. U. van Rienen, *Numerical Methods in Computational Electrodynamics.* Linear Systems in Practical Applications.
13. B. Engquist, L. Johnsson, M. Hammill, F. Short (eds.), *Simulation and Visualization on the Grid.*
14. E. Dick, K. Riemslagh, J. Vierendeels (eds.), *Multigrid Methods VI.*
15. A. Frommer, T. Lippert, B. Medeke, K. Schilling (eds.), *Numerical Challenges in Lattice Quantum Chromodynamics.*
16. J. Lang, *Adaptive Multilevel Solution of Nonlinear Parabolic PDE Systems.* Theory, Algorithm, and Applications.
17. B.I. Wohlmuth, *Discretization Methods and Iterative Solvers Based on Domain Decomposition.*
18. U. van Rienen, M. Günther, D. Hecht (eds.), *Scientific Computing in Electrical Engineering.*
19. I. Babuška, P.G. Ciarlet, T. Miyoshi (eds.), *Mathematical Modeling and Numerical Simulation in Continuum Mechanics.*
20. T.J. Barth, T. Chan, R. Haimes (eds.), *Multiscale and Multiresolution Methods.* Theory and Applications.
21. M. Breuer, F. Durst, C. Zenger (eds.), *High Performance Scientific and Engineering Computing.*
22. K. Urban, *Wavelets in Numerical Simulation.* Problem Adapted Construction and Applications.
23. L.F. Pavarino, A. Toselli (eds.), *Recent Developments in Domain Decomposition Methods.*

24. T. Schlick, H.H. Gan (eds.), *Computational Methods for Macromolecules: Challenges and Applications.*

25. T.J. Barth, H. Deconinck (eds.), *Error Estimation and Adaptive Discretization Methods in Computational Fluid Dynamics*.

26. M. Griebel, M.A. Schweitzer (eds.), *Meshfree Methods for Partial Differential Equations*.

27. S. Müller, *Adaptive Multiscale Schemes for Conservation Laws*.

28. C. Carstensen, S. Funken, W. Hackbusch, R.H.W. Hoppe, P. Monk (eds.), *Computational Electromagnetics*.

29. M.A. Schweitzer, *A Parallel Multilevel Partition of Unity Method for Elliptic Partial Differential Equations*.

30. T. Biegler, O. Ghattas, M. Heinkenschloss, B. van Bloemen Waanders (eds.), *Large-Scale PDE-Constrained Optimization*.

31. M. Ainsworth, P. Davies, D. Duncan, P. Martin, B. Rynne (eds.), *Topics in Computational Wave Propagation*. Direct and Inverse Problems.

32. H. Emmerich, B. Nestler, M. Schreckenberg (eds.), *Interface and Transport Dynamics.* Computational Modelling.

33. H.P. Langtangen, A. Tveito (eds.), *Advanced Topics in Computational Partial Differential Equations.* Numerical Methods and Diffpack Programming.

34. V. John, *Large Eddy Simulation of Turbulent Incompressible Flows.* Analytical and Numerical Results for a Class of LES Models.

35. E. Bänsch (ed.), *Challenges in Scientific Computing - CISC 2002.*

36. B.N. Khoromskij, G. Wittum, *Numerical Solution of Elliptic Differential Equations by Reduction to the Interface.*

37. A. Iske, *Multiresolution Methods in Scattered Data Modelling.*

38. S.-I. Niculescu, K. Gu (eds.), *Advances in Time-Delay Systems.*

39. S. Attinger, P. Koumoutsakos (eds.), *Multiscale Modelling and Simulation.*

40. R. Kornhuber, R. Hoppe, J. Périaux, O. Pironneau, O. Wildlund, J. Xu (eds.), *Domain Decomposition Methods in Science and Engineering.*

41. T. Plewa, T. Linde, V.G. Weirs (eds.), *Adaptive Mesh Refinement – Theory and Applications.*

42. A. Schmidt, K.G. Siebert, *Design of Adaptive Finite Element Software.* The Finite Element Toolbox ALBERTA.

43. M. Griebel, M.A. Schweitzer (eds.), *Meshfree Methods for Partial Differential Equations II.*

44. B. Engquist, P. Lötstedt, O. Runborg (eds.), *Multiscale Methods in Science and Engineering.*

45. P. Benner, V. Mehrmann, D.C. Sorensen (eds.), *Dimension Reduction of Large-Scale Systems.*

46. D. Kressner, *Numerical Methods for General and Structured Eigenvalue Problems.*

47. A. Boriçi, A. Frommer, B. Joó, A. Kennedy, B. Pendleton (eds.), *QCD and Numerical Analysis III*.

48. F. Graziani (ed.), *Computational Methods in Transport.*

49. B. Leimkuhler, C. Chipot, R. Elber, A. Laaksonen, A. Mark, T. Schlick, C. Schütte, R. Skeel (eds.), *New Algorithms for Macromolecular Simulation.*

50. M. Bücker, G. Corliss, P. Hovland, U. Naumann, B. Norris (eds.), *Automatic Differentiation: Applications, Theory, and Implementations.*

51. A.M. Bruaset, A. Tveito (eds.), *Numerical Solution of Partial Differential Equations on Parallel Computers.*

52. K.H. Hoffmann, A. Meyer (eds.), *Parallel Algorithms and Cluster Computing.*

53. H.-J. Bungartz, M. Schäfer (eds.), *Fluid-Structure Interaction.*

54. J. Behrens, *Adaptive Atmospheric Modeling.*

55. O. Widlund, D. Keyes (eds.), *Domain Decomposition Methods in Science and Engineering XVI.*

56. S. Kassinos, C. Langer, G. Iaccarino, P. Moin (eds.), *Complex Effects in Large Eddy Simulations.*

57. M. Griebel, M.A Schweitzer (eds.), *Meshfree Methods for Partial Differential Equations III.*

58. A.N. Gorban, B. Kégl, D.C. Wunsch, A. Zinovyev (eds.), *Principal Manifolds for Data Visualization and Dimension Reduction.*

59. H. Ammari (ed.), *Modeling and Computations in Electromagnetics: A Volume Dedicated to Jean-Claude Nédélec.*

60. U. Langer, M. Discacciati, D. Keyes, O. Widlund, W. Zulehner (eds.), *Domain Decomposition Methods in Science and Engineering XVII.*

61. T. Mathew, *Domain Decomposition Methods for the Numerical Solution of Partial Differential Equations.*

62. F. Graziani (ed.), *Computational Methods in Transport: Verification and Validation.*

63. M. Bebendorf, *Hierarchical Matrices.* A Means to Efficiently Solve Elliptic Boundary Value Problems.

64. C.H. Bischof, H.M. Bücker, P. Hovland, U. Naumann, J. Utke (eds.), *Advances in Automatic Differentiation.*

65. M. Griebel, M.A. Schweitzer (eds.), *Meshfree Methods for Partial Differential Equations IV*.

66. B. Engquist, P. Lötstedt, O. Runborg (eds.), *Multiscale Modeling and Simulation in Science.*

67. I.H. Tuncer, Ü. Gülcat, D.R. Emerson, K. Matsuno (eds.), *Parallel Computational Fluid Dynamics 2007.*

68. S. Yip, T. Diaz de la Rubia (eds.), *Scientific Modeling and Simulations*.

69. A. Hegarty, N. Kopteva, E. O'Riordan, M. Stynes (eds.), *BAIL* 2008 – *Boundary and Interior Layers.*

70. M. Bercovier, M.J. Gander, R. Kornhuber, O. Widlund (eds.), *Domain Decomposition Methods in Science and Engineering XVIII.*

71. B. Koren, C. Vuik (eds.), *Advanced Computational Methods in Science and Engineering.*

72. M. Peters (ed.), *Computational Fluid Dynamics for Sport Simulation.*

73. H.-J. Bungartz, M. Mehl, M. Schäfer (eds.), *Fluid Structure Interaction II - Modelling, Simulation, Optimization.*

74. D. Tromeur-Dervout, G. Brenner, D.R. Emerson, J. Erhel (eds.), *Parallel Computational Fluid Dynamics 2008.*

75. A.N. Gorban, D. Roose (eds.), *Coping with Complexity: Model Reduction and Data Analysis.*

76. J.S. Hesthaven, E.M. Rønquist (eds.), *Spectral and High Order Methods for Partial Differential Equations.*

77. M. Holtz, *Sparse Grid Quadrature in High Dimensions with Applications in Finance and Insurance*.

78. Y. Huang, R. Kornhuber, O.Widlund, J. Xu (eds.), *Domain Decomposition Methods in Science and Engineering XIX*.

79. M. Griebel, M.A. Schweitzer (eds.), *Meshfree Methods for Partial Differential Equations V*.

80. P.H. Lauritzen, C. Jablonowski, M.A. Taylor, R.D. Nair (eds.), *Numerical Techniques for Global Atmospheric Models*.

81. C. Clavero, J.L. Gracia, F.J. Lisbona (eds.), *BAIL 2010 – Boundary and Interior Layers, Computational and Asymptotic Methods*.

82. B. Engquist, O. Runborg, Y.R. Tsai (eds.), *Numerical Analysis and Multiscale Computations*.

83. I.G. Graham, T.Y. Hou, O. Lakkis, R. Scheichl (eds.), *Numerical Analysis of Multiscale Problems*.

84. A. Logg, K.-A. Mardal, G. Wells (eds.), *Automated Solution of Differential Equations by the Finite Element Method*.

85. J. Blowey, M. Jensen (eds.), *Frontiers in Numerical Analysis - Durham 2010*.

86. O. Kolditz, U.-J. Gorke, H. Shao, W. Wang (eds.), *Thermo-Hydro-Mechanical-Chemical Processes in Fractured Porous Media - Benchmarks and Examples*.

87. S. Forth, P. Hovland, E. Phipps, J. Utke, A. Walther (eds.), *Recent Advances in Algorithmic Differentiation*.

88. J. Garcke, M. Griebel (eds.), *Sparse Grids and Applications*.

89. M. Griebel, M.A. Schweitzer (eds.), *Meshfree Methods for Partial Differential Equations VI*.

90. C. Pechstein, *Finite and Boundary Element Tearing and Interconnecting Solvers for Multiscale Problems*.

91. R. Bank, M. Holst, O. Widlund, J. Xu (eds.), *Domain Decomposition Methods in Science and Engineering XX*.

92. H. Bijl, D. Lucor, S. Mishra, C. Schwab (eds.), *Uncertainty Quantification in Computational Fluid Dynamics*.

93. M. Bader, H.-J. Bungartz, T. Weinzierl (eds.), *Advanced Computing*.

94. M. Ehrhardt, T. Koprucki (eds.), *Advanced Mathematical Models and Numerical Techniques for Multi-Band Effective Mass Approximations*.

95. M. Azaïez, H. El Fekih, J.S. Hesthaven (eds.), *Spectral and High Order Methods for Partial Differential Equations ICOSAHOM 2012*.

96. M.P. Desjarlais, F. Graziani, R. Redmer, S.B. Trickey (eds.), *Frontiers and Challenges in Warm Dense Matter*.

97. J. Garcke, D. Pflüger (eds.), *Sparse Grids and Applications – Munich 2012*.

98. J. Erhel, M. Gander, L. Halpern, G. Pichot, T. Sassi, O. Widlund (eds.), *Domain Decomposition Methods in Science and Engineering XXI*.

99. R. Abgrall, H. Beaugendre, P.M. Congedo, C. Dobrzynski, V. Perrier, M. Ricchiuto (eds.), *High Order Nonlinear Numerical Methods for Evolutionary PDEs - HONOM 2013*.

For further information on these books please have a look at our mathematics catalogue at the following URL: www.springer.com/series/3527

Monographs in Computational Science and Engineering

1. J. Sundnes, G.T. Lines, X. Cai, B.F. Nielsen, K.-A. Mardal, A. Tveito, *Computing the Electrical Activity in the Heart.*

For further information on this book, please have a look at our mathematics catalogue at the following URL: www.springer.com/series/7417

Texts in Computational Science and Engineering

1. H. P. Langtangen, *Computational Partial Differential Equations.* Numerical Methods and Diffpack Programming. 2nd Edition
2. A. Quarteroni, F. Saleri, P. Gervasio, *Scientific Computing with MATLAB and Octave.* 4th Edition
3. H. P. Langtangen, *Python Scripting for Computational Science*. 3rd Edition
4. H. Gardner, G. Manduchi, *Design Patterns for e-Science*.
5. M. Griebel, S. Knapek, G. Zumbusch, *Numerical Simulation in Molecular Dynamics*.
6. H. P. Langtangen, *A Primer on Scientific Programming with Python.* 4th Edition
7. A. Tveito, H. P. Langtangen, B. F. Nielsen, X. Cai, *Elements of Scientific Computing.*
8. B. Gustafsson, *Fundamentals of Scientific Computing.*
9. M. Bader, *Space-Filling Curves.*
10. M. Larson, F. Bengzon, *The Finite Element Method: Theory, Implementation and Applications.*
11. W. Gander, M. Gander, F. Kwok, *Scientific Computing: An Introduction using Maple and MATLAB.*

For further information on these books please have a look at our mathematics catalogue at the following URL: www.springer.com/series/5151