



Science Arts & Métiers (SAM)

is an open access repository that collects the work of Arts et Métiers Institute of Technology researchers and makes it freely available over the web where possible.

This is an author-deposited version published in: <https://sam.ensam.eu>
Handle ID: <http://hdl.handle.net/10985/15515>

To cite this version :

Grégoire PONT, Pierre BRENNER, Paola CINNELLA, Bruno MAUGARS, Jean-Christophe ROBINET - Multiple-correction hybrid k -exact schemes for high-order compressible RANS-LES simulations on fully unstructured grids - Journal of Computational Physics - Vol. 350, p.45-83 - 2017

Any correspondence concerning this service should be sent to the repository

Administrator : scienceouverte@ensam.eu



Multiple-correction hybrid k -exact schemes for high-order compressible RANS-LES simulations on fully unstructured grids

Grégoire Pont^{a,b,*}, Pierre Brenner^a, Paola Cinnella^b, Bruno Maugars^b, Jean-Christophe Robinet^b

^a Airbus Safran Launchers, Les Mureaux, France

^b DynFluid Laboratory, Arts et Metiers ParisTech, 151 Boulevard de l'hopital, 75013 Paris, France

A B S T R A C T

A Godunov's type unstructured finite volume method suitable for highly compressible turbulent scale-resolving simulations around complex geometries is constructed by using a successive correction technique. First, a family of k -exact Godunov schemes is developed by recursively correcting the truncation error of the piecewise polynomial representation of the primitive variables. The keystone of the proposed approach is a quasi-Green gradient operator which ensures consistency on general meshes. In addition, a high-order single-point quadrature formula, based on high-order approximations of the successive derivatives of the solution, is developed for flux integration along cell faces. The proposed family of schemes is compact in the algorithmic sense, since it only involves communications between direct neighbors of the mesh cells. The numerical properties of the schemes up to fifth-order are investigated, with focus on their resolvability in terms of number of mesh points required to resolve a given wavelength accurately. Afterwards, in the aim of achieving the best possible trade-off between accuracy, computational cost and robustness in view of industrial flow computations, we focus more specifically on the third-order accurate scheme of the family, and modify locally its numerical flux in order to reduce the amount of numerical dissipation in vortex-dominated regions. This is achieved by switching from the upwind scheme, mostly applied in highly compressible regions, to a fourth-order centered one in vortex-dominated regions. An analytical switch function based on the local grid Reynolds number is adopted in order to warrant numerical stability of the recentering process. Numerical applications demonstrate the accuracy and robustness of the proposed methodology for compressible scale-resolving computations. In particular, supersonic RANS/LES computations of the flow over a cavity are presented to show the capability of the scheme to predict flows with shocks, vortical structures and complex geometries.

Keywords:

High order finite volume
 k -Exact
Vortex-centered scheme
Primitive variable reconstruction
Recentering process
General grid
Hybrid RANS/LES

1. Introduction

The development of high-order schemes is a very active research field in Computational Fluid Dynamics. Such schemes have the potential of reducing the computational cost associated with achieving a desired level of accuracy for a numerical simulation, since their truncation error decreases more rapidly than that of a lower-order method, at least for sufficiently smooth flow fields. Besides their improved truncation error properties, valid in the limit of vanishing mesh size, higher order methods are generally more efficient in terms of resolvability, i.e. the ability to represent accurately a given mode of the solution on a finite computational grid (see, e.g. [1–4]). The better resolvability properties of high-order methods are particularly useful for flow simulations involving a wide range of spatial and temporal scales, like Direct Numerical Simulations (DNS) and Large Eddy Simulations (LES) of turbulent flows [5–7,2–4].

The present work aims in particular at developing an accurate and efficient methodology for compressible flow computations dominated by large turbulent structures like those caused by, e.g., massive boundary layer separation. Typical examples are provided by flows past launchers and missiles, propulsive devices (engine chambers and nozzles...), and reentry bodies. These flows represent a challenge for numerical simulation methods since models based on the Reynolds-averaged equations (RANS) are not able to provide an accurate representation of the unsteady turbulent structures. On the other hand, DNS and even LES are still much too expensive for high-Reynolds flows about complex industrial configurations [8]. Recently, hybrid modeling approaches combining RANS and LES [9–12], referred-to in the following as HRL (Hybrid RANS-LES) models, have come into interest in the attempt to provide a tradeoff between computational cost and accuracy. HRL approaches share the common property of resolving the larger scales of turbulent flow whenever the flow dynamics is dominated by such scales (for instance in recirculating regions and shear layers) and the computational grid is fine enough to resolve them correctly, while applying RANS modeling elsewhere. A review of hybrid HRL models is beyond the scope of this paper. The interested reader may refer to, e.g. [13,14]. Instead, we focus on the development of a high-order numerical strategy well suited for HRL simulations on unstructured grids, and we investigate the influence of the numerical properties of the resolution scheme on HRL simulations. Precisely we look for a numerical approach that fulfills the following requirements:

- it resolves the relevant flow scales accurately by using the smallest number of cells;
- it handles accurately and robustly compressible flows with strong shock waves;
- it can be applied to arbitrarily complex geometrical configurations;
- it ensures high accuracy on fully unstructured grids, including highly distorted ones.

Reviews about high-order schemes for structured and unstructured grids and high-order methods in general can be found in [15,16]. In particular, a significant amount of research has been carried out in the last 20 years about Discontinuous Galerkin (DG), Spectral Volume or Spectral Difference methods [6,7,16]. In these kind of methods, extra degrees of freedom are introduced in each cell to fit a high-order polynomial to the solution. In spite of the significant advancements achieved recently (see, e.g. [17] for a review), such methods still encounter difficulties for highly compressible flow simulations. Moreover, they cannot be easily implemented into existing production codes used in industry, and the development of new production codes to take advantage of these new high-order schemes requires considerable implementation, verification and validation efforts.

For all these reasons, in this work we restrain our attention to higher-order finite volume methods (FVM), which do not introduce internal degrees of freedom and can be easily integrated within existing industrial solvers. Additionally, FVM generally possess good shock capturing properties. More particularly, the methodology investigated in this work falls within the family of Godunov methods [18]. Higher-order accuracy is achieved by locally reconstructing the solution within each mesh cell [19] by means of polynomials, the coefficients of which are related to the successive derivatives of the solution. This provides at any time an accurate representation of the solution which is a piecewise polynomial of the same degree k than the one used for reconstruction. This is why such methods are usually referred-to as k -exact schemes. The foundations of this approach can be found in the work of Barth and Frederickson [20,21] and Vandersbilck and Deconinck [22]. More recent work can be found in [23–27]. In particular, several concepts further developed in the present paper can be found in [28,25]. The main stumbling block for the development of k -exact approximations is represented by the calculation and storage of the successive derivatives of the solution, which may be particularly challenging for unstructured grids and parallel computing, since it requires accessing nonlocal data [24,25]. This is generally done by using least-mean square (LMS) reconstructions [23–26] or, more rarely, by means of the successive application of Gauss–Green type approximations of the gradients [27]. Unfortunately, LMS reconstructions can be very problematic for highly stretched and curved grids like those used, e.g., in RANS calculations of boundary layers. Moreover, LMS involves a large amount of extra storage [23,29]. A thorough discussion of the limitations of LMS gradient approximations and some possible cures are presented in [30,31,26]. The main problems are related to the ill-conditioning of the LMS problem for highly irregular grids, leading to the appearance of extremely large approximation errors. Conditioning issues become more and more significant as the scheme order is increased, and for 3D problems [26]. Significant errors are also introduced by Gauss–Green gradient approximations when used in conjunction with high-aspect-ratio, curved grids [30,31] but they are less critical than for LMS. Another difficult point is the necessity of high-order quadrature formulae for the approximation of flux integrals. This is generally done by introducing Gauss quadrature points on each interface, albeit this requires the reconstruction and storage of a significant amount of information, especially for the higher orders. Additionally, since FVM generally use average values

of the conservative variables on each cell as the problem unknowns, an efficient deconvolution process is required to reconstruct pointwise values of the primitive variables (used by robust compressible schemes) from the cell averages of these conservative variables. Finally, a suitable numerical flux must be used at cell interfaces, since values reconstructed from the right and from the left of the interface may be discontinuous. Upwind methods are often used for this purpose, since they introduce an intrinsic numerical dissipation that damps non-physical oscillations and ensures stability for the solution of hyperbolic problems. Unfortunately, this dissipation also damps the smallest scales of motion, leading to poor resolvability for the LES of turbulent flow [32] and, as a consequence, in LES regions of RANS/LES simulations. Note that in the numerical simulation of turbulent flows, there are three sources of damping: the first one is represented by the molecular viscosity ν , the second one is the eddy viscosity ν_t introduced by the turbulence or subgrid model, and the last one is the numerical viscosity ν_n introduced by the numerical scheme in use. The second kind of viscosity is equal to zero for DNS and for the so called “implicit” LES approaches (see e.g. [33]), while it generally dominates the other two when RANS models are considered. The main idea behind HRL models is to lower the turbulent viscosity ν_t of an underlying RANS model and to resolve directly part of the turbulent structures in regions selected according to both grid resolution and physical criteria. In the resolved part of the turbulent spectrum, the numerical viscosity must be negligible compared to eddy viscosity to avoid spurious damping of turbulent structures. On the other hand, in complex configurations that exhibit strong shocks and fine turbulent structures simultaneously, the flow solver should be able to capture the former robustly, while providing an accurate representation of the latter. Dispersion errors should also be quantified carefully.

In the aim of overcoming the above mentioned drawbacks of upwind k -exact methods, in this paper we first present a general procedure for constructing high-order k -exact schemes based on recursive corrections of the approximate successive derivatives of flow variables, as also done in [25]. Unlike Ref. [25], in this work we make no use of LMS approximations. Instead, we construct the required derivatives starting from a Gauss–Green type formulation and by successively correcting the truncation error terms. The successive procedure requires only exchanges between direct neighboring cells at each correction step and, as such, is computationally compact, even if the final stencil of the scheme is not compact. A similar correction approach has been recently proposed in [27], which however ensures truly high accuracy only on regular Cartesian grids. In this work, the Gauss–Green-based approximations of the derivatives are corrected to ensure consistency on arbitrary unstructured grids, including highly irregular ones. The proposed successive correction algorithm can be efficiently parallelized as shown in [25]. In order to reduce storage and computational requirements, the flux integrals are not approximated by means of Gauss quadrature formulae. Instead, an efficient one-point quadrature formula involving the approximated successive derivatives already used for solution reconstruction is introduced, which minimizes the truncation error on general skew interfaces.

The numerical properties of the proposed family of k -exact schemes are then analyzed analytically for a model equation, both in terms of local truncation errors and of spectral properties. Specifically, the behavior of the dispersion and dissipation errors is investigated for k -exact schemes up to the fifth-order of accuracy, and the number of points per wavelengths required to achieve some target error levels is compared to the scheme operation count. It is shown that, for three-dimensional problems, the operation count and memory requirements grow rapidly with the scheme order, while grid resolution requirements improve more slowly. The third-order scheme of the family, characterized by sufficiently low dispersion errors for unsteady aerodynamics applications, is then selected for the following of the study, devoted to the development of a strategy for improving the scheme resolvability in terms of numerical dissipation properties. In particular, a hybrid centered/upwind discretization is introduced (based on both physical considerations and numerical stability criteria), which ensures the optimal amount of numerical dissipation in both vortex- and shock-dominated flow regions. Hybrid schemes were also investigated in [34,35]. Here, we develop an hybrid 3rd-order upwind/4th-order centered scheme by modifying the numerical flux of a third-order k -exact scheme. The baseline 3rd-order k -exact upwind scheme is locally recentered in vortex-dominated regions under a stability condition based on the grid Reynolds number and a vortex sensor. As a result, the scheme is converted locally into a 4th-order accurate centered scheme that does not damp the turbulent structures, while introducing the minimal amount of numerical dissipation required to avoid spurious oscillations in the rest of the computational domain.

Because it is based on the use of successive corrections both for solution reconstruction and for flux approximation, by distorted-grid corrections, and by a hybrid numerical dissipation term, the proposed approach is christened a “multiple-correction” hybrid scheme.

The proposed scheme has been implemented in FLUSEPA,¹ the unstructured finite-volume solver developed by Airbus Safran Launchers company to calculate compressible, multidimensional, unsteady, viscous and reactive flow over bodies in relative motion. The scheme is first assessed in detail for simple, well documented two-dimensional test cases, to demonstrate its capability of preserving high accuracy on highly deformed grids and the effectiveness of the hybrid centered/upwind strategy. Specifically, the well-known Ringleb flow, for which an analytical solution of the compressible Euler equations is available, is solved on computational grids with various topologies (quadrangular, triangular, hybrid, and with overlapping/patched grids). The vortex advection and the shock/vortex interaction problems are then used to investigate the dissipation properties of the corrected scheme. This is subsequently applied to the hybrid RANS/LES computation of the transonic flow over a square cylinder. A complex supersonic flow past a surface-mounted cylinder is also simulated to

demonstrate the capabilities of the proposed approach for supersonic flows characterized by strong shocks. Lastly, a final application involving the M219 cavity exact wind tunnel model is presented to demonstrate the ability of the proposed approach of handling complex geometries.

2. Governing equations

We look for numerical solutions of the compressible Reynolds-averaged or filtered Navier–Stokes equations, supplemented by a turbulence or subgrid model, respectively. These can be formally written as a system of conservation laws of the form:

$$\frac{d}{dt} \iiint_{\Omega_{CV}} \mathbf{w} d\Omega + \oint_{A_{CV}} \mathbf{F} \cdot \mathbf{n} dS \quad (1)$$

where Ω_{CV} is a fixed control volume with boundary A_{CV} , \mathbf{n} is the outer unit normal, \mathbf{w} is the conservative variable vector:

$$\mathbf{w} = (\rho, \rho \mathbf{u}, \rho E)^T \quad (2)$$

and \mathbf{F} is the flux density:

$$\mathbf{F} = \mathbf{F}_E + \mathbf{F}_V \quad (3)$$

where the inviscid flux \mathbf{F}_E is given by:

$$\mathbf{F}_E = (\rho \mathbf{u}, \rho \mathbf{u} \otimes \mathbf{u} + P \underline{\underline{I}}, \rho \mathbf{u} H)^T \quad (4)$$

and the viscous flux \mathbf{F}_V is

$$\mathbf{F}_V = (0, \underline{\underline{\tau}} + \underline{\underline{\tau}}_t, (\underline{\underline{\tau}} + \underline{\underline{\tau}}_t) \cdot \mathbf{u} - \mathbf{q} - \mathbf{q}_t)^T \quad (5)$$

In Equations (4) and (5), ρ is the density, \mathbf{u} the velocity vector, P the pressure, E the specific total energy, H the specific total enthalpy, $\underline{\underline{\tau}}$ the viscous stress tensor and \mathbf{q} the heat flux vector. In RANS/LES regions of the computational domain, the preceding flow properties represent averaged/filtered quantities, respectively. From a more general viewpoint, \mathbf{w} represents the resolved part of the flow, i.e. the larger scales of motion captured by the calculation, which eventually reduce to the mere average field for full RANS. In addition, the Reynolds/subgrid stress tensor $\underline{\underline{\tau}}_t$, and the turbulent heat flux \mathbf{q}_t , are introduced to model the effect of the non-resolved part of the flow on the resolved one. In the following, $\underline{\underline{\tau}}_t$ is described by using an eddy viscosity (Boussinesq) approximation, and is supposed to be linearly related to the average/filtered velocity gradient via a turbulent viscosity coefficient ν_t . A turbulent Fourier law with a constant turbulent Prandtl number is used to model \mathbf{q}_t . The turbulent viscosity ν_t is determined by means of an automatic HRL model. Precisely, for the present computations we use the Delayed Detached Eddy Simulation model (DDES) of [36], improved by introducing Chauvet's turbulence length scale [37]. Precisely, the present implementation of model is based on a compressible version of the Spalart–Allmaras one equation model, proposed by Catris and Aupoix [38] (see [39] for more applications).

3. Numerical method

System (1) is approximated by means of a finite volume methodology on unstructured grids. The computational domain Ω_{CV} is partitioned in N cells Ω_J with boundary A_J , such that:

$$\Omega_{CV} = \bigcup_{J=1}^N \Omega_J \quad \text{and} \quad A_J = \bigcup_{K=1}^P A_{JK} \quad (6)$$

where index K denotes the K th neighbor to cell Ω_J , denoted Ω_K , and A_{JK} the interface between two neighbors:

$$A_{JK} = \Omega_J \cap \Omega_K. \quad (7)$$

In each cell Ω_J , and for any field Ψ , we introduce the space average over one cell $\bar{\Psi}_J$ and the pointwise value at the cell center \mathbf{x}_j of Ω_J , denoted Ψ_j (see Fig. 1). They are defined, respectively:

$$\bar{\Psi}_J = \frac{1}{|\Omega_J|} \iiint_{\Omega_J} \Psi d\Omega, \quad \text{and} \quad \Psi_j = \Psi(\mathbf{x}_j), \quad \text{with} \quad \mathbf{x}_j = \frac{1}{|\Omega_J|} \iiint_{\Omega_J} \mathbf{x} d\Omega \quad (8)$$

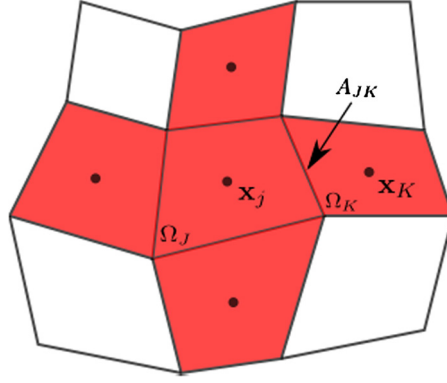


Fig. 1. Sketch of a 2D mesh and of the first neighborhood $s_1(J)$ of a mesh cell Ω_J .

In the above, $|\Omega_J| = \iiint_{\Omega_J} d\Omega$ denotes the volume of Ω_J . In the present approach, the problem unknowns are represented by the average volume of the conservative variables over each cell, i.e, the values:

$$\bar{\mathbf{w}}_J = \frac{1}{|\Omega_J|} \iiint_{\Omega_J} \mathbf{w} d\Omega. \quad (9)$$

Let us introduce the m th order volume moment of Ω_J , $\mathcal{M}_J^{(m)}$, defined as:

$$\mathcal{M}_J^{(m)} = \frac{1}{|\Omega_J|} \iiint_{\Omega_J} (\mathbf{x} - \mathbf{x}_J)^{\otimes m} d\Omega, \quad (10)$$

where, for any vector \mathbf{v} :

$$\mathbf{v}^{\otimes m} = \underbrace{\mathbf{v} \otimes \mathbf{v} \otimes \dots \otimes \mathbf{v}}_{m \text{ times}} \quad (11)$$

with \otimes the dyadic product between two vectors. Note that $\mathcal{M}^{(0)}$ is equal to the scalar unity and $\mathcal{M}^{(1)}$ is a null vector by definition of the cell center \mathbf{x}_j .

We also introduce $\mathcal{S}_{A_{JK}}^{(m)}$, the m th-order moment of A_{JK} , defined as:

$$\mathcal{S}_{A_{JK}}^{(m)} = \iint_{A_{JK}} (\mathbf{x} - \mathbf{x}_J)^{\otimes m} \cdot \mathbf{n} dS \quad (12)$$

Specifically, $\mathcal{S}_{A_{JK}}^{(0)}$ is the oriented surface of A_{JK} :

$$\mathcal{S}_{A_{JK}}^{(0)} = \iint_{A_{JK}} \mathbf{n} dS \quad (13)$$

With the preceding notations, the system of conservation laws referred to a computational cell writes:

$$|\Omega_J| \frac{d\bar{\mathbf{w}}_J}{dt} + \sum_{K=1}^P \iint_{A_{JK}} \mathbf{F} \cdot \mathbf{n} dS = 0 \quad (14)$$

where P is the number of faces of cell Ω_J .

To complete the discretization, we need to find an approximation for the surface integral in (14) (flux term) as a function of the problem unknowns $\bar{\mathbf{w}}_J$. For this purpose we use a k -exact reconstruction technique, in conjunction with an efficient successive correction algorithm to achieve orders of accuracy greater than two.

For each grid cell Ω_J let us consider a neighborhood $s(J)$ made by the union of cell Ω_J and of a set of surrounding cells (see Fig. 1). Specifically, we call $s_1(J)$ the set made of the current cell plus its first neighbors, i.e. cells that share at least a face with Ω_J , s_2 the set made of s_1 plus the second neighbors, etc.

Definition 3.1 (*k-exact function*). A function ${}^k\Psi$ is said to be k -exact over $s(J)$ if its restriction to $s(J)$, denoted ${}^k\Psi|_{s(J)}$ is a polynomial of degree k . In other words, it exists a polynomial of degree k , denoted ${}^k\mathcal{P}$ such that:

$${}^k\Psi|_{s(J)} = {}^k\mathcal{P}|_{s(J)}$$

Now, let us call

$$\overline{\Psi} = \{\overline{\Psi}_J; J \in s(J)\}$$

the collection of the cell-averages of Ψ over the cells of $s(J)$.

Definition 3.2 (*k-exact differentiation operator*). A linear operator $\mathcal{D}^{(n)}$ is called a k -exact approximate differentiation operator over $s(J)$ if, for any m -exact function ${}^m\Psi$, with $m \leq k$, then:

$$({}^k\mathcal{D}^{(n)} {}^m\Psi)|_J = (\mathbf{D}^{(n)} {}^m\Psi)|_J$$

where $\mathbf{D}^{(n)}(\bullet)|_j$ represents the n th spatial tensor derivative of ${}^m\Psi$ at point \mathbf{x}_j .

Let be r the order of accuracy of a differential operator, denoted by $\mathcal{D}_r^{(n)}$. Then, using a Taylor expansion, we can see that:

$$({}^k\mathcal{D}^{(n)} {}^m\Psi)|_J = (\mathbf{D}^{(n)} {}^m\Psi)|_J + \mathcal{O}(h^r), \quad r = k - n + 1$$

In order to simplify notations, we assume in the following:

$${}^k\mathcal{D}^{(n)}|_J = \mathcal{D}_{k-n+1}^{(n)}|_J$$

The truncation error of the first order accurate differentiation operator for the n th derivative is of the form:

$$\mathcal{D}_1^{(n)}|_J = \mathbf{D}^{(n)}|_J + \sum_{p=1}^m \mathcal{H}_{p+1}^{(n)}|_J \mathbf{D}^{(n+p)}|_J + \mathcal{O}(h^{(m+1)}) \quad (15)$$

This formula allows to find automatically any derivative at any order of accuracy combined with the successive corrections algorithm described in section 3.4. It will be shown later that the \mathcal{H} matrices can be obtained by applying a 1-exact differentiation operator (and specifically the so-called “quasi-Green” 1-exact operator of Eq. (3.2)) to specific grid metrics. For example, Eq. (57) gives the expression of the matrix $\mathcal{H}_2^{(1)}$.

In the following, we present a k -exact reconstruction of the solution within computational cells that can be used to find high-order approximations of Eq. (14). For robustness reasons, the reconstruction is applied to the primitive variables vector:

$$\mathbf{q} = (\mathbf{u}, P, T)^T, \quad \text{with} \quad \mathbf{q} = \mathbf{q}(\mathbf{w}) \quad (16)$$

Precisely, the primitive and conservative variables are related by:

$$\begin{cases} \mathbf{u} = (\rho\mathbf{u})/\rho \\ P = (\gamma - 1) \left((\rho E) - \frac{1}{2} \frac{|\rho\mathbf{u}|^2}{\rho} \right) \\ T = \frac{P}{\rho R} \end{cases}$$

where the last two equations hold for perfect gas with constant R and constant specific heat ratio γ . For further convenience, we introduce for each cell J the vector $\tilde{\mathbf{q}}_J = \tilde{\mathbf{q}}(\overline{\mathbf{w}}_J)$ defined as:

$$\tilde{\mathbf{q}}_J = (\tilde{\mathbf{u}}_J, \tilde{P}_J, \tilde{T}_J)^T \quad (17)$$

with:

$$\begin{cases} \tilde{\mathbf{u}}_J = (\overline{\rho\mathbf{u}})_J / \overline{\rho}_J \\ \tilde{P}_J = (\gamma - 1) \left((\overline{\rho E})_J - \frac{1}{2} \frac{|\overline{\rho\mathbf{u}}_J|^2}{\overline{\rho}_J} \right) \\ \tilde{T}_J = \frac{\tilde{P}_J}{R\overline{\rho}_J} \end{cases} \quad (18)$$

Note that quantities \tilde{q}_J are second-order-accurate approximations of the pointwise values of the primitive variables at cell center \mathbf{x}_j . For instance, from the definition of volume average we get:

$$\tilde{\mathbf{u}}_J = \frac{(\rho \mathbf{u})_j + \mathcal{O}(h^2)}{\rho_j + \mathcal{O}(h^2)} = \mathbf{u}_j + \mathcal{O}(h^2) \quad (19)$$

where h is a characteristic mesh size.

For the sake of clarity, in the following we first introduce the 2nd-order reconstruction procedure, then we generalize it to any order of accuracy by means of a successive correction procedure.

3.1. Second-order 1-exact approximation

A 2nd-order approximation of the flux integral in Eq. (14) is given by the classic second order formula:

$$\iint_{A_{JK}} \mathbf{F} \cdot \mathbf{n} dS = \mathbf{F}_\Gamma \cdot \mathcal{S}_{A_{JK}}^{(0)} + \mathbf{D}^{(1)} \mathbf{F} \Big|_\Gamma \cdot \mathcal{S}_{A_{JK}}^{(1)} + |A_{JK}| \mathcal{O}(h^2), \quad (20)$$

where \mathbf{F}_Γ is the physical flux \mathbf{F} at a point $\Gamma \in A_{JK}$, with $|A_{JK}| = |\mathcal{S}_{A_{JK}}^{(0)}|$. For a flat cell interface, \mathbf{n} is constant and Γ is the center of gravity of face A_{JK} , with coordinates:

$$\mathbf{x}_\Gamma = \frac{1}{|A_{JK}|} \iint_{A_{JK}} \mathbf{x} dS$$

With this choice, $\mathcal{S}_{A_{JK}}^{(1)} = 0$ and equation (21) reduces to:

$$\iint_{A_{JK}} \mathbf{F} \cdot \mathbf{n} dS = \mathbf{F}_\Gamma \cdot \mathcal{S}_{A_{JK}}^{(0)} + |A_{JK}| \mathcal{O}(h^2) \quad (21)$$

For a general skew grid, $\mathcal{S}_{A_{JK}}^{(1)} \neq 0$. In our case, Γ is chosen in such a way to minimize the first-order error term in (20), see [Appendix B](#).

To build the numerical flux \mathbf{F}_Γ , a second-order approximation of \mathbf{q} at point Γ is first constructed using information from cells at the left and at the right of face A_{JK} based on a linear reconstruction of the solution over Ω_J and Ω_K . Values of q reconstructed from the left and from the right are referred to as \mathbf{q}_{Γ_L} and \mathbf{q}_{Γ_R} , respectively. These are used as an input for a 1D Riemann solver applied to the face normal direction [18].

A linear second-order accurate approximation of \mathbf{q} over one cell is obtained by Taylor-expanding \mathbf{q} around the cell center up to first-order terms:

$$\mathbf{q}(\mathbf{x}) = \mathbf{q}_j + \mathbf{D}^{(1)} \mathbf{q} \Big|_j \cdot (\mathbf{x} - \mathbf{x}_j) + \mathcal{O}(h^2) \quad (22)$$

where $\mathbf{D}^{(1)} \mathbf{q} \Big|_j$ is the first order tensor derivative of \mathbf{q} at point j , i.e., its gradient. Afterwards, the pointwise values of \mathbf{q} and $\mathbf{D}^{(1)} \mathbf{q} \Big|_j$ are replaced by a second-order and a first order approximation, respectively.

A straightforward second-order approximation for \mathbf{q}_j is given by:

$$\mathbf{q}_j = \tilde{\mathbf{q}}_J + \mathcal{O}(h^2) \quad (23)$$

To approximate $\mathbf{D}^{(1)} \mathbf{q} \Big|_j$, we use a linear combination of the cell averages:

$$(\mathcal{D}^{(1)} \bar{\mathbf{q}}) \Big|_J = \sum_{K \in s(J)} \mathbf{W}_{JK} \bar{\mathbf{q}}_K \quad (24)$$

where $\mathbf{W}_{JK} = \mathcal{O}(\frac{1}{h})$ is a vector of weights associated to the K th cell of $s(J)$. Note that, if \mathbf{q} is a 1-exact function, then for all $\Omega_J \in \Omega_{CV}$:

$$\mathbf{D}^{(1)} \mathbf{q} = \boldsymbol{\alpha} \quad (25)$$

with $\boldsymbol{\alpha}$ a piecewise constant vector, and

$$\mathbf{D}^{(m)} \mathbf{q} = 0 \quad \forall m \geq 2 \quad (26)$$

In this case, by integration of Eq. (22):

$$\bar{\mathbf{q}}_K = \mathbf{q}_k, \quad (27)$$

and

$$(\mathcal{D}^{(1)}\bar{\mathbf{q}})|_J = \mathbf{D}^{(1)}\mathbf{q}|_j = \sum_{K \in \mathcal{S}(J)} \mathbf{W}_{JK} \mathbf{q}_k \quad (28)$$

The last term can be expanded in Taylor series around \mathbf{x}_j , yielding:

$$(\mathcal{D}^{(1)}\bar{\mathbf{q}})|_J = \sum_{K \in \mathcal{S}(J)} \mathbf{W}_{JK} \left(\mathbf{q}_j + \mathbf{D}^{(1)}\mathbf{q}|_j \cdot (\mathbf{x}_k - \mathbf{x}_j) + \mathcal{O}(h^2) \right) \quad (29)$$

where second-order terms are null because of the 1-exactness of \mathbf{q} . Rearranging terms,

$$(\mathcal{D}^{(1)}\bar{\mathbf{q}})|_J = \left(\sum_{K \in \mathcal{S}(I)} \mathbf{W}_{JK} \right) \mathbf{q}_j + \sum_{K \in \mathcal{S}(I)} \mathbf{W}_{JK} (\mathbf{D}^{(1)}(\mathbf{q})|_j \cdot (\mathbf{x}_k - \mathbf{x}_j)) \quad (30)$$

By using Eqs (30), (28) and (25) it is deduced that the 1-exactness conditions for the operator $\mathcal{D}^{(1)}$ are:

$$\sum_{K \in \mathcal{S}(I)} \mathbf{W}_{JK} = \mathbf{0}, \quad (0\text{-exactness condition}) \quad (31)$$

$$\sum_{K \in \mathcal{S}(I)} \mathbf{W}_{JK} (\boldsymbol{\alpha} \cdot (\mathbf{x}_k - \mathbf{x}_j)) = \boldsymbol{\alpha}, \quad \forall \boldsymbol{\alpha} \quad (32)$$

The operator $\mathcal{D}^{(1)}$ is second-order accurate in Cartesian grids but zero-order accurate in general grids. Note that an operator which respect the two previous conditions is called 1-exact. In the next section, we present how to construct an 1-exact operator $\mathcal{D}_1^{(1)}$. In practice, the operator $\mathcal{D}_1^{(1)}$ is not applied to $\bar{\mathbf{q}}_J$ but instead to $\tilde{\mathbf{q}}_J$, as $\bar{\mathbf{q}}$ is readily computed from the problem unknowns $\tilde{\mathbf{w}}$ by means of Eqs (18). Actually, since $\tilde{\mathbf{q}}_J = \bar{\mathbf{q}}_J + \mathcal{O}(h^2)$, then:

$$(\mathcal{D}_1^{(1)}\bar{\mathbf{q}})|_J = (\mathcal{D}_1^{(1)}\tilde{\mathbf{q}})|_J + \mathcal{O}(h) \quad (33)$$

So, to the first order of accuracy, using $\tilde{\mathbf{q}}$ in place of $\bar{\mathbf{q}}$ does not affect the leading error term.

In summary, the final expression for the gradient approximation operator $\mathbf{D}^{(1)}\mathbf{q}|_j$ is:

$$(\mathcal{D}_1^{(1)}\tilde{\mathbf{q}})|_J = \sum_{K \in \mathcal{S}(J)} \mathbf{W}_{JK} \tilde{\mathbf{q}}_k \quad (34)$$

The procedure for computing the weights \mathbf{W}_{JK} is described in the next Section.

With \mathbf{q}_j and $\mathbf{D}^{(1)}\mathbf{q}|_j$, approximated to the second and first order by means of equations (27) and (34), respectively, the left and right values of \mathbf{q} at point Γ are computed as:

$$\begin{cases} \mathbf{q}_{\Gamma_L} = \mathbf{q}_j + (\mathcal{D}_1^{(1)}\tilde{\mathbf{q}})|_j \cdot (\mathbf{x}_\Gamma - \mathbf{x}_j) + \mathcal{O}(h^2) \\ \mathbf{q}_{\Gamma_R} = \mathbf{q}_k + (\mathcal{D}_1^{(1)}\tilde{\mathbf{q}})|_k \cdot (\mathbf{x}_\Gamma - \mathbf{x}_k) + \mathcal{O}(h^2) \end{cases}$$

where j and k denote the centers of cells J and K , respectively. A Riemann solver is finally used to calculate the numerical flux \mathbf{F}_Γ from \mathbf{q}_{Γ_L} and \mathbf{q}_{Γ_R} . Without the application of a limiting procedure, the scheme is not stable near discontinuities. Limiters are applied to the reconstructed left and right primitive variables in order to enforce monotonicity constraints [40]. Detailed studies about general stability of the scheme on any grids are beyond the scope of the present work.

3.2. Gradient computation on general meshes: simple correction method

We have seen in the preceding Section that a 1-exact differentiation operator applied to the primitive variables, $\mathcal{D}_1^{(1)}\mathbf{q}|_J$ is of the form (34). We still have to determine the coefficients W_{JK} . There are several approaches for gradient computation on unstructured grids like Green-Gauss or Weighted-Least-Squares [27,24,31].

In this work, the coefficients \mathbf{W}_{JK} are determined by means of a quasi-Green formulation supplemented by a correction step to recover consistency on general meshes [40]. The correction step is essential to preserve consistency on arbitrary unstructured grid.

According to the Gauss–Green theorem, the exact gradient $\mathbf{D}^{(1)}$ satisfies:

$$\iint_{\Omega_J} \mathbf{D}^{(1)} \mathbf{q} d\Omega = \iint_{\partial\Omega_J} \mathbf{q} \cdot \mathbf{n} dS = \sum_{K \in \mathcal{S}(J)} \iint_{A_{JK}} \mathbf{q} \cdot \mathbf{n} dS, \quad (35)$$

Using a second-order approximation of the volume integral, the cell-center gradient $\mathbf{D}^{(1)} \mathbf{q}|_J$

$$|\Omega_J| \mathbf{D}^{(1)} \mathbf{q}|_J = \sum_{K \in \mathcal{S}(J)} \iint_{A_{JK}} \mathbf{q} \cdot \mathbf{n} dS + |\Omega_J| \mathcal{O}(h^2) \quad (36)$$

The integrals on the right-hand side can be approximated using a one-point-per-face integration scheme:

$$\iint_{A_{JK}} \mathbf{q} \cdot \mathbf{n} dS \simeq (\beta_K \bar{\mathbf{q}}_K + (1 - \beta_K) \bar{\mathbf{q}}_J) \cdot \mathbf{A}_{JK}, \quad (37)$$

where \mathbf{A}_{JK} is the oriented surface pointing from J to K , and β_K is a weight associated to cell K . Coefficients β_K are weights based on the relative distance of cell centers \mathbf{x}_k to face A_{JK} :

$$\beta_K = \frac{\|\mathbf{x}_\Gamma - \mathbf{x}_j\|}{\|\mathbf{x}_\Gamma - \mathbf{x}_j\| + \|\mathbf{x}_k - \mathbf{x}_\Gamma\|}$$

Using (37) to compute the gradient from (36) corresponds to using equation (34) with weight vectors given by:

$$\mathbf{V}_{JK} = \frac{1}{|\Omega_J|} \beta_K \mathbf{A}_{JK} \quad \text{if } J \neq K, \quad \mathbf{V}_{JJ} = \frac{1}{|\Omega_J|} \sum_{K \in \mathcal{S}(J)} (1 - \beta_K) \mathbf{A}_{JK} \quad (38)$$

The previous weight vectors always satisfies the first conditions of (31), which is the 0-exactness condition, but not the second one, which corresponds to the 1-exactness condition.

Since the approximation operator (34) is a linear application, it can be represented by a matrix \mathbf{M}_1 such that:

$$\sum_{K \in \mathcal{S}(J)} \mathbf{V}_{JK} (\boldsymbol{\alpha} \cdot (\mathbf{x}_k - \mathbf{x}_j)) = \mathbf{M}_1 \boldsymbol{\alpha}, \quad \forall \boldsymbol{\alpha} \quad (39)$$

For a 1-exact function ${}^1\Psi$:

$$\bar{{}^1\Psi}_K - \bar{{}^1\Psi}_J = \mathbf{D}^{(1)1} \Psi|_j \cdot (\mathbf{x}_k - \mathbf{x}_j), \quad (40)$$

multiplying by $\beta_K \mathbf{A}_{JK}$ and summing over K :

$$\sum_{K \in \mathcal{S}(J)} \beta_K (\bar{{}^1\Psi}_K - \bar{{}^1\Psi}_J) \mathbf{A}_{JK} = \sum_{K \in \mathcal{S}(J)} \beta_K \left(\mathbf{D}^{(1)1} \Psi|_j \cdot (\mathbf{x}_k - \mathbf{x}_j) \right) \mathbf{A}_{JK} \quad (41)$$

$$\Leftrightarrow \sum_{K \in \mathcal{S}(J)} \left(\beta_K \bar{{}^1\Psi}_K + (1 - \beta_K) \bar{{}^1\Psi}_J \right) \mathbf{A}_{JK} = \sum_{K \in \mathcal{S}(J)} \left(\beta_K (\bar{{}^1\Psi}_K - \bar{{}^1\Psi}_J) \right) \mathbf{A}_{JK} \quad (42)$$

since cell faces form a closed surface.

Combining equations (36), (37), (39), and (42), we finally get:

$$\mathbf{D}^{(1)1} \Psi|_j = \mathbf{M}_1^{-1} \sum_{K \in \mathcal{S}(J)} \left(\beta_K \bar{{}^1\Psi}_K + (1 - \beta_K) \bar{{}^1\Psi}_J \right) \mathbf{A}_{JK} \quad (43)$$

which is the quasi-Green gradient approximation. For a generic function Ψ , the gradient approximation $\mathcal{D}_1^{(1)}$ given by:

$$(\mathcal{D}_1^{(1)} \bar{\Psi})|_j = \mathbf{M}_1^{-1} \sum_{K \in \mathcal{S}(J)} \left(\beta_K \bar{\Psi}_K + (1 - \beta_K) \bar{\Psi}_J \right) \mathbf{A}_{JK} \quad (44)$$

is by construction 1-exact according to Definition 3.2, and as such it is at least first-order accurate for any function Ψ and any general mesh.

The matrix \mathbf{M}_1 is called hereafter the “simple correction” matrix, and is given by:

$$\mathbf{M}_1 = \sum_{K \in \mathcal{S}(J)} \beta_K (\mathbf{x}_k - \mathbf{x}_j) \otimes \mathbf{A}_{JK} \quad (45)$$

Note that for regular Cartesian grid, $\mathbf{M}_1^{-1} = \frac{\mathbb{I}}{|\Omega_J|}$: thus, the role of \mathbf{M}_1^{-1} is correcting the gradient approximation to enforce first-order accuracy on any grid.

3.3. Higher-order k -exact approximations

Higher-order approximations of the fluxes and, consequently, higher-order schemes, can be obtained by using a polynomial reconstruction of higher degree for the primitive variables, along with high-order integration formula for the flux terms in (14). Unlike previous work [28] that uses costly Gauss quadrature formulas, here we adopt a one-point integration formula based on high-accurate approximations of the flux and its derivatives at a point Γ of the cell interface.

In the following, we first introduce the one-point integration formula, then we illustrate the high-order reconstruction procedure over each cell. Finally, we derive approximations of the flux and its successive derivatives at point Γ from the surrounding cell-centered values.

3.3.1. One-point integration formula for fluxes

The construction of higher-order schemes starts with the development of a high-order accurate approximation formula for the flux term:

$$\iint_{A_{JK}} \mathbf{F} \cdot \mathbf{n} dS, \quad (46)$$

to replace equation (21) that is only second-order accurate. This is achieved by expanding \mathbf{F} in Taylor series at a suitable integration point Γ of A_{JK} . Plugging this expression into (46) gives:

$$\iint_{A_{JK}} \mathbf{F} \cdot \mathbf{n} dS = \sum_{m=0}^{n-1} \frac{1}{m!} \mathbf{D}^{(m)} \mathbf{F}|_{\Gamma} \cdot \mathcal{S}_{A_{JK}}^{(m)} + |A_{JK}| \mathcal{O}(h^n) \quad (47)$$

where $\mathbf{D}^{(l)} \mathbf{F}|_{\Gamma}$ is the l th-tensor derivative of \mathbf{F} at point Γ , and h a characteristic grid size. We recall that:

$$\mathbf{D}^{(0)} \mathbf{F}|_{\Gamma} = \mathbf{F}|_{\Gamma}, \quad \mathbf{D}^{(1)} \mathbf{F}|_{\Gamma} = \mathbf{grad}(\mathbf{F})|_{\Gamma} \quad \text{and} \quad \mathcal{S}_{A_{JK}}^{(0)} = \iint_{A_{JK}} \mathbf{n} dS \quad (48)$$

Note that, in the very specific cases where the interfaces are flat, Γ is the face center and $\mathcal{S}_{A_{JK}}^{(1)} = 0$. In the general case, the Γ point is selected as the point that minimizes the first-order error term (see Section 3.1 and Appendix B).

To achieve a n th-order approximation of the surface integral, the flux and its derivatives at point Γ have to be reconstructed from cell-centered averages with suitable accuracy: precisely, $\mathbf{D}^{(m)} \mathbf{F}|_{\Gamma}$ has to be reconstructed at order $n - m$. For instance, to achieve a third-order accuracy ($n = 3$), $\mathbf{F}|_{\Gamma}$ has to be reconstructed at order 3, $\mathbf{D}^{(1)} \mathbf{F}|_{\Gamma}$ at order 2 and $\mathbf{D}^{(2)} \mathbf{F}|_{\Gamma}$ at order 1. For this purpose, we choose to rewrite the m th derivatives of the flux in terms of the m th derivatives of the primitive variables $\mathbf{D}^{(m)} \mathbf{q}$. In turn, these are evaluated by means of a high-order k -exact reconstruction, along with the successive correction algorithm of Section 3.4. This change of evaluation point involves a negligible additional cost, since approximations of the derivatives $\mathbf{D}^{(m)} \mathbf{q}$ are already used in other steps of the method.

3.3.2. High-order k -exact reconstruction procedure

To calculate approximations of the solution at cell faces, high-order polynomials are constructed over each cell by using again Taylor-series expansions. These are obtained by including higher-order terms in Eq. (22):

$$\mathbf{q}(\mathbf{x}) = \mathbf{q}_j + \mathbf{D}^{(1)} \mathbf{q}|_j \cdot (\mathbf{x} - \mathbf{x}_j) + \sum_{m=2}^{n-1} \mathbf{D}^{(m)} \mathbf{q}|_j \cdot \frac{(\mathbf{x} - \mathbf{x}_j)^{\otimes m}}{m!} + \mathcal{O}(h^n) \quad (49)$$

A n th-order reconstruction of the primitive variables is obtained by truncating (49) at order $n - 1$, and by approximating \mathbf{q}_j , $\mathbf{D}^{(1)} \mathbf{q}|_j$, and $\mathbf{D}^{(m)} \mathbf{q}|_j$, $m = 2, \dots, n - 1$ at order n , $n - 1$, and $n - m$, respectively.

First, a high-order accurate deconvolution procedure is developed to compute pointwise values of the variables at cell centers from the problem unknowns. Similarly to the 2nd-order scheme of Section 3.1, the polynomial reconstruction is applied to the primitive variables.

By Taylor-series expanding the primitive variables \mathbf{q} around the cell center \mathbf{x}_j , and by averaging over the cell, one gets:

$$\mathbf{q}_j = \bar{\mathbf{q}}_j - \sum_{m=2}^{n-1} \frac{1}{m!} \mathcal{M}_j^{(m)} : \mathbf{D}^{(m)} \mathbf{q}|_j + \mathcal{O}(h^n) \quad (50)$$

where $\mathcal{M}_j^{(m)}$ is the m th volume moment of Ω_j , given by Equation (10), $\mathbf{D}^{(m)} \mathbf{q}|_j$ is the m th-order tensor derivative of \mathbf{q} at point \mathbf{x}_j and $:$ denotes the inner product between two m th-order tensors. Note that the contribution of first-order

derivatives (i.e. the term $m = 1$ in the sum) vanishes because first-order moments with respect to the volume center of Ω_J are identically null.

An approximation of \mathbf{q}_j at order $n > 2$ requires to find an $(n - m)$ th-order accurate approximation for the m th derivative $\mathbf{D}^{(m)}\mathbf{q}|_j$ as well as an n th-order approximation for $\bar{\mathbf{q}}_J$.

Both $\bar{\mathbf{q}}_J$ and $\mathbf{D}^{(m)}\mathbf{q}|_j$ have to be computed from the problem unknowns $\bar{\mathbf{w}}_J$, for which purpose we use the procedure described in the following.

As a first step, we start writing an approximation for $\bar{\mathbf{q}}_J$. We know that:

$$\bar{\mathbf{q}}_J = \tilde{\mathbf{q}}_J + \mathcal{O}(h^2), \quad (51)$$

where $\tilde{\mathbf{q}}_J$ is directly related to the problem unknowns $\bar{\mathbf{w}}_J$ through equations (18). To achieve a higher-order approximation for $\bar{\mathbf{q}}_J$, we need to correct the second-order error term.

Precisely, we look for a relation of the form:

$$\bar{\mathbf{q}}_J = \tilde{\mathbf{q}}_J + \Delta\mathbf{q} + \mathcal{O}(h^n). \quad (52)$$

The correction term $\Delta\mathbf{q}$ depends on the specific primitive variable under consideration (\mathbf{u} , P or T). The full expressions are given in Appendix A. For the sake of brevity, hereafter we describe only the general principles of the procedure.

First, $\tilde{\mathbf{q}}_J = \tilde{\mathbf{q}}(\bar{\mathbf{w}}_J)$ is Taylor-series expanded in order to express the average value of conservative variables $\bar{\mathbf{w}}_J$ in term of their pointwise values \mathbf{w}_j and of their successive derivatives at point \mathbf{x}_j , $\mathbf{D}^{(m)}\mathbf{w}|_j$:

$$\tilde{\mathbf{q}}_J = \tilde{\mathbf{q}}(\bar{\mathbf{w}}_J) = \tilde{\mathbf{q}}\left(\mathbf{w}_j + \frac{1}{|\Omega_J|} \sum_{m=2}^{n-1} \frac{1}{m!} \mathcal{M}_J^{(m)} : \mathbf{D}^{(m)}\mathbf{w}|_j + \mathcal{O}(h^n)\right) \quad (53)$$

In turn, Eq. (53) is Taylor-series expanded around \mathbf{w}_j , and its derivatives are re-written in terms of the conservatives variables, to get:

$$\tilde{\mathbf{q}}_J = \mathbf{q}_j + \delta\mathbf{q}\left(\mathbf{D}^{(2)}\mathbf{w}|_j, \mathbf{D}^{(3)}\mathbf{w}|_j, \dots, \mathbf{D}^{(n-1)}\mathbf{w}|_j\right) \quad (54)$$

where $\delta\mathbf{q}$ represents again a correction term, which depends on $\mathbf{D}^{(m)}\mathbf{w}|_j$, $m = 2, \dots, n - 1$, allowing to evaluate \mathbf{q}_j from $\tilde{\mathbf{q}}_J$ to an order of accuracy greater than 2.

Substracting equation (54) from equation (50) and re-arranging terms, one finally obtains:

$$\Delta\mathbf{q} = \bar{\mathbf{q}}_J - \tilde{\mathbf{q}}_J + \mathcal{O}(h^n) \quad (55)$$

The important thing to note here is that, unlike equation (54), $\Delta\mathbf{q}$ depends only on derivatives of \mathbf{q} at point \mathbf{x}_j up to the $(n - 2)$ th order, and not the $(n - 1)$ th since the error terms depending on the $(n - 1)$ th derivatives cancel (see Appendix A for more details). Moreover, getting an n th-order approximation for $\bar{\mathbf{q}}_J$ requires only an $(n - m - 1)$ th order approximation for $\mathbf{D}^{(m)}\mathbf{q}|_j$.

The required approximations are constructed by means of the successive corrections algorithm described in the following Section.

3.4. Method of successive corrections

The higher-order approximation of the derivatives $\mathbf{D}^{(m)}\mathbf{q}|_j$ and of the cell average \bar{q}_J required for the k -exact reconstruction and flux integration are obtained by means of an iterative procedure, based on successive corrections of the truncation error terms.

At the beginning of the procedure, a first-order approximation of the first derivatives at point \mathbf{x}_j is obtained by applying the 1-exact operator $\mathcal{D}_1^{(1)}$ of Section 3.2 to $\tilde{\mathbf{q}}$ (which is a second-order approximation for $\bar{\mathbf{q}}$).

The preceding first-order approximation allows to construct a third-order approximation for $\bar{\mathbf{q}}_J$ through the correction step (55).

Then, the first-order differentiation operator can be re-applied to this higher-order approximation of $\bar{\mathbf{q}}$. The associated truncation error is of the form:

$$\mathcal{E}_2^{(1)} = (\mathcal{D}_1^{(1)}\bar{\mathbf{q}})|_j - \mathbf{D}^{(1)}\mathbf{q}|_j = \mathbf{D}^{(2)}\mathbf{q}|_j \cdot \mathcal{H}_2^{(1)} + \mathcal{O}(h^2) \quad (56)$$

where $\mathcal{H}_2^{(1)}$ is a matrix that depends only on the geometrical properties of the mesh in use, the elements of which are $\mathcal{O}(h)$. By Taylor-series expansion of the $\mathcal{D}_1^{(1)}$ operator and by application of the 1-exactness condition, it is easily seen that:

$$\mathcal{H}_2^{(1)} = \mathbf{M}_1^{-1} \sum_{K \in \mathcal{S}(J)} \mathbf{W}_{JK} (\mathbf{x}_k - \mathbf{x}_j)^{\otimes 2} = \mathcal{D}_1^{(1)}(\overline{\mathbf{x} \otimes \mathbf{x}}|_J) \quad (57)$$

An approximation of the 2nd derivatives at point \mathbf{x}_j , $\mathbf{D}^{(2)}\mathbf{q}|_j$, is obtained by applying $\mathcal{D}_1^{(1)}$ twice to $\bar{\mathbf{q}}_j$. This leads, on general meshes, to an inconsistent operator:

$$\left(\mathcal{D}_1^{(1)}\left(\mathcal{D}_1^{(1)}\bar{\mathbf{q}}\right)\right)|_j = \mathbf{D}^{(2)}\mathbf{q}|_j + O(1)$$

Similarly to the procedure used in Section 3.2 to construct $\mathcal{D}_1^{(1)}$, a 2-exact (hence, consistent) approximation for the second derivatives is obtained by setting:

$$\left(\mathcal{D}_1^{(1)}\left(\mathcal{D}_1^{(1)}\bar{\mathbf{q}}\right)\right)|_j = \mathbf{M}_2\left(\mathcal{D}_1^{(2)}\bar{\mathbf{q}}\right)|_j \quad (58)$$

where \mathbf{M}_2 is a linear application such that:

$$\left(\mathcal{D}_1^{(2)}\bar{\mathbf{q}}\right)|_j = \mathbf{M}_2^{-1}\left(\mathcal{D}_1^{(1)}\left(\mathcal{D}_1^{(1)}\bar{\mathbf{q}}\right)\right)|_j = \mathbf{D}^{(2)}\mathbf{q}|_j + O(h)$$

It can be shown that:

$$\mathbf{M}_2 = \mathcal{D}_1^{(1)}\mathcal{D}_1^{(1)}(\overline{\mathbf{x}\otimes\mathbf{x}}|_j)$$

Details about the construction of \mathbf{M}_2 are given in [28].

The first-order approximation of the second-order derivatives can be used to correct the truncation error of $\mathcal{D}_1^{(1)}$ in Eq. (56), leading to a second-order approximation for the first derivatives:

$$\left(\mathcal{D}_2^{(1)}\bar{\mathbf{q}}\right)|_j = \mathbf{D}^{(1)}\mathbf{q}|_j + O(h^2) \quad (59)$$

Finally, the 3rd-order approximation of $\bar{\mathbf{q}}_j$, the second-order approximation of $\mathbf{D}^{(1)}\mathbf{q}|_j$, and the first-order approximation of $\mathbf{D}^{(2)}\mathbf{q}|_j$ can be used to build a 3rd-order reconstruction formula for \mathbf{q} over the mesh cell Ω_J , according to Eq. (49).

The preceding procedure can be repeated iteratively to further increase the order of accuracy:

- The second-order approximation of $\mathbf{D}^{(1)}|_j$ and the first order approximation of $\mathbf{D}^{(2)}|_j$ can be used to correct the preceding third-order approximation of $\bar{\mathbf{q}}_j$ (Eq. (51)), thus obtaining a fourth-order approximation.
- The second-order operator for the first derivatives and the first-order operator for the second derivatives are applied to the fourth-order approximation of $\bar{\mathbf{q}}_j$.
- An approximation for the third-order derivatives is obtained by applying $\mathcal{D}_1^{(1)}$ to $\left(\mathcal{D}_1^{(2)}\bar{\mathbf{q}}\right)|_j$.
- The resulting (inconsistent) third differentiation operator is corrected via a suitable linear application, \mathbf{M}_3 , leading to a 1-exact operator, denoted $\mathcal{D}_1^{(3)}$. Note that $\mathbf{M}_3 = \mathcal{D}_1^{(1)}\mathcal{D}_1^{(2)}(\overline{\mathbf{x}\otimes\otimes\mathbf{x}}|_j)$.
- The first-order approximation of the third derivatives is used to correct the first-order truncation error term of $\mathcal{D}_1^{(2)}$, of the form:

$$\varepsilon_2^{(2)} = \left(\mathcal{D}_1^{(2)}\bar{\mathbf{q}}\right)|_j - \mathbf{D}^{(2)}\mathbf{q}|_j = \mathbf{D}^{(3)}\mathbf{q}|_j \cdot \mathcal{H}_2^{(2)} + O(h^2) \quad (60)$$

where $\mathcal{H}_2^{(2)} = \mathcal{D}_1^{(1)}\mathcal{D}_1^{(1)}(\overline{\mathbf{x}\otimes\otimes\mathbf{x}}|_j)$ is again a matrix that depends only on the geometrical properties of the mesh in use, the elements of which are $O(h)$. After correction, a 2nd-order accurate, second differentiation operator is obtained, denoted $\mathcal{D}_2^{(2)}$.

- Similarly, $\mathcal{D}_1^{(3)}$ and $\mathcal{D}_2^{(2)}$ are used to correct the second-order truncation error term of $\mathcal{D}_2^{(1)}$, leading to a 3rd-order first differentiation operator, $\mathcal{D}_3^{(1)}$:

$$\varepsilon_3^{(1)} = \left(\mathcal{D}_1^{(1)}\bar{\mathbf{q}}\right)|_j - \mathbf{D}^{(1)}\mathbf{q}|_j = \mathbf{D}^{(2)}\mathbf{q}|_j \cdot \mathcal{H}_2^{(1)} + \mathbf{D}^{(3)}\mathbf{q}|_j \cdot \mathcal{H}_3^{(1)} + O(h^3) \quad (61)$$

where $\mathcal{H}_3^{(1)} = \mathcal{D}_1^{(1)}(\overline{\mathbf{x}\otimes\otimes\mathbf{x}}|_j)$.

- The higher-order approximations of $\bar{\mathbf{q}}_j$ and of the derivatives are used finally to build a 4th order reconstruction of \mathbf{q} over a cell.

In a more general way, an n -exact approximation of \mathbf{q} can be obtained through the following:

Algorithm [Successive correction algorithm]

1. Initialization: compute $\tilde{\mathbf{q}}_J = \bar{\mathbf{q}}(\bar{w}_J)$ from Eq. (18)
2. Set $\bar{\mathbf{q}}_J \leftarrow \tilde{\mathbf{q}}_J$

(* with an error $O(h^2)$ *)

3. Set $(\mathcal{D}_1^{(1)}\bar{\mathbf{q}})|_J \leftarrow (\mathcal{D}_1^{(1)}\tilde{\mathbf{q}})|_J$

(* with an error $O(h)$ *)

4. **for** $m \leftarrow 1$ **to** n

5. **do** Use $(\mathcal{D}_m^{(1)}\bar{\mathbf{q}})|_J, (\mathcal{D}_{m-1}^{(2)}\bar{\mathbf{q}})|_J, \dots, (\mathcal{D}_1^{(m-1)}\bar{\mathbf{q}})|_J$ to compute

6. $\bar{\mathbf{q}}_J = \tilde{\mathbf{q}}_J + \Delta\mathbf{q} + O(h^{m+2})$

7. **for** $k \leftarrow 1$ **to** m

8. Set $(\mathcal{D}_{m-k+1}^{(k)}\bar{\mathbf{q}})|_J \leftarrow (\mathcal{D}_{m-k+1}^{(k)}\bar{\mathbf{q}})|_J$

(* Differentiate the highest-order estimate available for $\bar{\mathbf{q}}$ using the highest-order available differentiation operator *)

9. Compute

10. $(\mathcal{D}_1^{(1)}(\mathcal{D}_1^{(m)}\bar{\mathbf{q}}))|_J = \mathbf{D}^{(m+1)}\mathbf{q}|_J + O(1)$

11. Compute

12. $(\mathcal{D}_1^{(m+1)}\bar{\mathbf{q}})|_J = \mathbf{M}_{m+1}^{-1}(\mathcal{D}_1^{(1)}(\mathcal{D}_1^{(m)}\bar{\mathbf{q}}))|_J = \mathbf{D}^{(m+1)}\mathbf{q}|_J + O(h)$

13. **for** $l \leftarrow m$ **to** 1

14. **do** Use $(\mathcal{D}_1^{(m+1)}\bar{\mathbf{q}})|_J, (\mathcal{D}_2^{(m)}\bar{\mathbf{q}})|_J, \dots, (\mathcal{D}_{m-l+1}^{(l+1)}\bar{\mathbf{q}})|_J$ to

15. correct the truncation error of $(\mathcal{D}_{m-l+1}^{(l)}\bar{\mathbf{q}})|_J$ and

16. obtain $(\mathcal{D}_{m-l+2}^{(l)}\bar{\mathbf{q}})|_J = \mathbf{D}^{(l)}\mathbf{q}|_J + O(h^{m-l+1})$

17. **return** the $m+2$ reconstruction of \mathbf{q} at a point \mathbf{x} ,

$$\mathbf{q}(\mathbf{x}) = \mathbf{q}_j + \sum_{m=1}^{n-1} (\mathcal{D}_{n-m}^{(m)}\bar{\mathbf{q}})|_J \cdot (\mathbf{x} - \mathbf{x}_j)^{\otimes m}$$

Note that, from a numerical point of view, the numerical cross derivatives do not verify the Schwarz theorem to machine accuracy, especially on irregular grids. To overcome this issue, we perform an average of numerical cross derivatives calculated in each direction.

The whole successive correction strategy relies on the recursive application of the 1-exact gradient operator and of geometrical correction matrices. Since such operator requires only information from the nearest cell neighbors, the correction procedure comes out to be compact in a computational sense. Of course, this does not mean that the approximation schemes obtained at the end of the procedure is compact. For example, the 2-exact scheme requires gradient information from cell neighbors, i.e. the stencil involves the first and second neighborhoods. More generally, increasing the accuracy by one order adds one more neighborhood to the discretization stencil.

For solid boundaries where the stencil is modified in the normal direction by suppressing the lacking neighboring cell from the gradient operator. With this approach, the formal order of accuracy of the scheme is not modified, even if the overall error level tends to increase. An in-depth analysis of the effect of the boundary conditions is warranted as future research.

3.5. Integration of the numerical flux

In Section 3.3.1, we presented a n th-order one-point formula (48) for flux integration along a cell interface. The formula involves a n th-order approximation of the flux density at the integration point Γ , $\mathbf{F}|_\Gamma$, as well as $(n-m)$ th-order approximations of its m th derivatives.

To calculate $\mathbf{F}|_\Gamma$, k -exact reconstructions of the primitive variable vector \mathbf{q} over cells at the left and at the right of the interface are used to approximate \mathbf{q} at point Γ to the n th order of accuracy from the left and from the right, respectively:

$$\begin{cases} \mathbf{q}_{\Gamma_L} = \mathbf{q}_j + \sum_{m=1}^{n-1} (\mathcal{D}_{n-m}^{(m)}\bar{\mathbf{q}})|_J \cdot \frac{(\mathbf{x}_\Gamma - \mathbf{x}_j)^{\otimes m}}{m!} + \mathcal{O}(h^n) \\ \mathbf{q}_{\Gamma_R} = \mathbf{q}_k + \sum_{m=1}^{n-1} (\mathcal{D}_{n-m}^{(m)}\bar{\mathbf{q}})|_K \cdot \frac{(\mathbf{x}_\Gamma - \mathbf{x}_k)^{\otimes m}}{m!} + \mathcal{O}(h^n) \end{cases}$$

where the indices j and k denote the centers of cells J and K , respectively; $(m-n)$ th-accurate differentiation operators $\mathcal{D}_{n-m}^{(m)}$ are constructed via the successive correction algorithm. Similarly to the second-order scheme described in 3.3.1, a donor-cell method [41] is used to chose the left or right value.

Once the reconstruction of the left and right values has been completed, a slope limiter is eventually applied to ensure monotonicity, an then the 1D exact Riemann solver is used to calculate the numerical flux \mathbf{F}_Γ from \mathbf{q}_{Γ_L} and \mathbf{q}_{Γ_R} .

The next step is to reconstruct the corrective terms at the right-hand side of equation (48). For this purpose, we need to estimate derivatives of the flux at point Γ , by using information from the right and from the left of the interface. Precisely,

polynomial reconstructions of the l th derivative at the $(n-l)$ th-order of accuracy are written by using approximations of the derivatives at the left and right cell centers, previously computed by using the successive correction algorithm:

$$\begin{cases} \left(\mathcal{D}_{n-l}^{(l)} \bar{\mathbf{q}} \right) \Big|_{\Gamma_L} = \left(\mathcal{D}_{n-l}^{(l)} \bar{\mathbf{q}} \right) \Big|_J + \sum_{m=l+1}^{n-l-1} \left(\mathcal{D}_{n-l-m}^{(m)} \bar{\mathbf{q}} \right) \Big|_J \cdot \frac{(\mathbf{x}_\Gamma - \mathbf{x}_J)^{\otimes m}}{m!} + \mathcal{O}(h^{n-l}) \\ \left(\mathcal{D}_{n-l}^{(l)} \bar{\mathbf{q}} \right) \Big|_{\Gamma_R} = \left(\mathcal{D}_{n-l}^{(l)} \bar{\mathbf{q}} \right) \Big|_K + \sum_{m=l+1}^{n-l-1} \left(\mathcal{D}_{n-l-m}^{(m)} \bar{\mathbf{q}} \right) \Big|_K \cdot \frac{(\mathbf{x}_\Gamma - \mathbf{x}_K)^{\otimes m}}{m!} + \mathcal{O}(h^{n-l}) \end{cases} \quad (62)$$

for any $l = 1, \dots, n-1$.

For any given cell face the same formula are used to approximate the numerical flux and the flux derivatives on both sides, thus leading to an intrinsically conservative approximation scheme.

4. Numerical properties

The numerical properties of the preceding family of numerical schemes are investigated for a simple 1D problem, namely, the linear advection equation:

$$\frac{\partial \alpha}{\partial t} + a \mathbf{D}^{(1)} \alpha = 0 \quad (63)$$

with $\mathbf{D}^{(1)} \alpha = \frac{\partial \alpha}{\partial x}$ for a 1D scalar problem, and $a \in \mathfrak{R}$. Equation (63) is discretized on a regular grid with space step h , so that cell centers are located at abscissas $x_j = ih$ and cell faces at $x_{j \pm \frac{1}{2}} = (j \pm \frac{1}{2})h$. By integration over one cell, (63) becomes:

$$h \frac{\partial \bar{\alpha}_J}{\partial t} + a(\alpha_{j+\frac{1}{2}} - \alpha_{j-\frac{1}{2}}) = 0 \quad (64)$$

where $\bar{\alpha}_J$ is the average value of α over the control volume $\Omega_J = [x_{j-\frac{1}{2}}, x_{j+\frac{1}{2}}]$:

$$\bar{\alpha}_J = \frac{1}{h} \int_{x_{j-\frac{1}{2}}}^{x_{j+\frac{1}{2}}} \alpha dx$$

To complete the numerical discretization, an approximation of $\alpha_{j \pm \frac{1}{2}}$ is constructed from cell averages over the neighboring cells, by means of the upwind method described in the preceding Section. For simplicity, assume $a > 0$: in this case, $\alpha_{j-\frac{1}{2}}$ and $\alpha_{j+\frac{1}{2}}$ are reconstructed at order n by using information from cell $j-1$ and i , respectively. We denote these upwind approximations as $\alpha_{L|j \pm \frac{1}{2}}$, so that (64) rewrites:

$$\frac{\partial \bar{\alpha}_J}{\partial t} + \frac{a}{h} \mathcal{L}_k(\bar{\alpha}) \Big|_J = 0 \quad (65)$$

with $\mathcal{L}_k(\bar{\alpha}) \Big|_J = \left(\alpha_{L|j+\frac{1}{2}} - \alpha_{L|j-\frac{1}{2}} \right)$ and k the order of the k -exact reconstruction in use. In the following analysis, we consider approximations up to the fifth-order of accuracy. For the present 1D linear problem, the k -exact reconstruction algorithm finally leads to the following approximations of the flux difference, for orders ranging from 2 to 5, respectively:

$$\mathcal{L}_2(\bar{\alpha}) \Big|_J = \frac{1}{4} (\bar{\alpha}_{j+1} + 3\bar{\alpha}_j - 5\bar{\alpha}_{j-1} + \bar{\alpha}_{j-2}) \quad (66)$$

$$\mathcal{L}_3(\bar{\alpha}) \Big|_J = \frac{1}{32} (\bar{\alpha}_{j+2} + 7\bar{\alpha}_{j+1} + 22\bar{\alpha}_j - 38\bar{\alpha}_{j-1} + 9\bar{\alpha}_{j-2} - 1\bar{\alpha}_{j-3}) \quad (67)$$

$$\mathcal{L}_4(\bar{\alpha}) \Big|_J = \frac{1}{384} (-4\bar{\alpha}_{j+3} + 15\bar{\alpha}_{j+2} + 96\bar{\alpha}_{j+1} + 255\bar{\alpha}_j - 468\bar{\alpha}_{j-1} \quad (68)$$

$$+ 117\bar{\alpha}_{j-2} - 8\bar{\alpha}_{j-3} - 3\bar{\alpha}_{j-4}) \quad (69)$$

$$\mathcal{L}_5(\bar{\alpha}) \Big|_J = \frac{1}{6144} (-17\bar{\alpha}_{j+4} - 47\bar{\alpha}_{j+3} + 324\bar{\alpha}_{j+2} + 1468\bar{\alpha}_{j+1} + 3930\bar{\alpha}_j \quad (70)$$

$$- 7386\bar{\alpha}_{j-1} + 1988\bar{\alpha}_{j-2} - 196\bar{\alpha}_{j-3} - 81\bar{\alpha}_{j-4} + 17\bar{\alpha}_{j-5}) \quad (71)$$

4.1. Truncation error analysis

To investigate the numerical properties of the preceding approximation, we rewrite it in terms of cell center pointwise values α_j . For this purpose, cell averaged values in (65) (with \mathcal{L}_k given by one of the operators (66)–(71)) are replaced by the following development

$$\alpha_j = \bar{\alpha}_j - \frac{h^2}{24} \mathbf{D}^{(2)}\alpha \Big|_j - \frac{h^4}{1920} \mathbf{D}^{(4)}\alpha \Big|_j + \mathcal{O}(h^6). \quad (72)$$

After some algebra, the truncation errors \mathcal{E} associated to the upwind k -exact schemes up to the 5th order of accuracy are written as:

$$\mathcal{E}_{\mathcal{L}_2} = -\frac{ah^2}{24} \mathbf{D}^{(3)}\alpha \Big|_i + \frac{ah^3}{8} \mathbf{D}^{(4)}\alpha \Big|_i + \mathcal{O}(h^4) \quad (73)$$

$$\mathcal{E}_{\mathcal{L}_3} = \frac{ah^3}{16} \mathbf{D}^{(4)}\alpha \Big|_i - \frac{199ah^4}{5760} \mathbf{D}^{(5)}\alpha \Big|_i + \mathcal{O}(h^5) \quad (74)$$

$$\mathcal{E}_{\mathcal{L}_4} = -\frac{19ah^4}{5760} \mathbf{D}^{(5)}\alpha \Big|_i - \frac{3ah^5}{64} \mathbf{D}^{(6)}\alpha \Big|_i + \mathcal{O}(h^6) \quad (75)$$

$$\mathcal{E}_{\mathcal{L}_5} = -\frac{19ah^5}{768} \mathbf{D}^{(6)}\alpha \Big|_i + \frac{17ah^6}{1008} \mathbf{D}^{(7)}\alpha \Big|_i + \mathcal{O}(h^7) \quad (76)$$

Inspection of the truncation errors (73) shows that odd-order schemes are dominant dissipative whereas even order scheme are dominant dispersive. Increasing the spatial order of accuracy from 2 to 3 leads to a reduction of the dispersive error (from $\frac{ah^2}{24} \mathbf{D}^{(3)} \Big|_i$ to $\frac{199ah^4}{5760} \mathbf{D}^{(5)} \Big|_i$), whereas the dissipation error remains of the same order, even if the error constant is twice smaller. A similar effect is obtained by increasing the order of accuracy from four to five, but the improvement in the dissipation error constant is slightly smaller (reduction by a factor 1.895). On the contrary, going from third to fourth order accuracy improves essentially the dissipation error (from $\frac{ah^3}{16} \mathbf{D}^{(4)} \Big|_i$ to $\frac{3ah^5}{64} \mathbf{D}^{(6)} \Big|_i$), whereas the dispersion error remains of the same order, although with a ten times smaller error constant.

4.2. Spectral properties

Truncation error analysis provides information on the behavior of the numerical schemes in the limit of vanishing mesh size. For finite grids, however, an analysis of the spectral properties of the spatial operator is useful to get insight on the numerical representation of a given solution wavelength.

Consider again the linear advection equation (63). The Fourier transform of (63) writes:

$$\widehat{\partial_t \alpha} = -i\hat{\alpha} \xi \frac{\widehat{\alpha}}{\Delta t} \quad (77)$$

with ξ the reduced wave number and $\hat{\alpha} = ah/\Delta t$ the CFL number. The right-hand side of the preceding equation is the exact transport operator in the spectral space. On the other hand, taking the Fourier transform of the discretized advection equation rewritten in terms of the pointwise values α gives:

$$\widehat{\partial_t \alpha} = -\hat{\alpha} \widehat{\mathcal{L}} \frac{\widehat{\alpha}}{\Delta t} \quad (78)$$

where $\mathcal{L} \in \mathbb{C}$ is the Fourier symbol of the spatial discretization.

By analogy with equation (77), we define a modified wavenumber $\xi^* = -i\widehat{\mathcal{L}}$, so that (78) rewrites:

$$\widehat{\partial_t \alpha} = -i\hat{\alpha} \xi^* \frac{\widehat{\alpha}}{\Delta t}$$

For an infinitely accurate scheme, $\xi^* = \xi$ (ξ being a pure real number). In practice, $\xi^* \neq \xi$ because of approximation errors. Specifically, the imaginary part of ξ^* is generally different from zero and it introduce a damping of the solution mode $\widehat{\alpha}$, whereas its real part govern the dispersion errors. To measure how accurately a numerical scheme represents a given wavenumber, we introduce the normalized dispersion error:

$$\mathcal{P}_\xi = \frac{|\mathcal{R}e(\xi^*) - \xi|}{\pi},$$

and the dissipation function:

$$\mathcal{A}_\xi = 1 - \exp(\mathcal{I}m(\xi^*)).$$

If $\xi^* = \xi$, then $\mathcal{P}_\xi = \mathcal{D}_\xi = 0$.

Table 1
Resolvability properties of various k -exact in terms of number of points per wavelength.

k -exactness	1	2	3	4
Order of accuracy	2	3	4	5
Points per wave length for dissipation	21	17	11	10
Points per wave length for dispersion	14	10	8	7

Table 2
Relative computational cost (compared to the 1-exact) of various k -exact schemes. The ranges for 2-exact to 4-exact schemes in the 2D and 3D cases depend on the way cross-derivatives are computed.

k -exactness	1	2	3	4
Order of accuracy	2	3	4	5
1D	1	2	3	4
2D	1	2.5÷3	4.5÷6	6÷10
3D	1	3÷4	6.3÷10	11.3÷20

Fig. 2 shows the dispersion (\mathcal{P}) and dissipation (\mathcal{A}) errors as functions of the modified wave number $\xi = k\Delta x$ for k -exact schemes ranging from order 2 to order 5. All of these schemes exhibit a significant dissipation error. For a more quantitative analysis, Table 1 provides the number of points per wave length necessary to achieve a target dissipation error, arbitrarily taken equal to 10^{-3} , a value often used in the literature (see e.g. [4]).

The table shows that increasing the order of accuracy does improve both dissipation and dispersion errors. Specifically, increasing the accuracy of even-order schemes (2 and 4) by one order reduces the number of points required to achieve the target dispersion error, whereas moving from 3rd- to 4th-order accuracy mostly improves resolution requirement for the target dissipation. For scale resolving compressible simulations of turbulent flows it is important to select numerical scheme with reasonably good dispersion properties. Nevertheless, dissipation plays a much more crucial role than dispersion (see for instance [42,43]). We notice that k -exact schemes of a given order have better resolvability properties in terms of dispersion compared to dissipation, due to their upwind character that leads to intrinsically dissipative schemes. Dispersion errors are reduced by about 40% when increasing the accuracy from second to third order, and decrease much more slowly when moving to higher-order schemes (gains of about 25% and about 14% when moving from order 3 to 4, and 4 to 5, respectively). Dissipation errors follow an almost similar trend. It appears then important to compare the expected gains in terms of resolvability properties to the over-cost introduced by the higher-order schemes. This is primarily due to the computation and storage of higher and higher-order derivatives of the solution during the recursive correction procedure. Table 2 provides estimates of the computational costs, in terms of operation count, for k -exact schemes of various orders in the one-, two- and three-dimensional cases. For each case, the operation count is normalized with respect to the 1-exact scheme. For higher-order k -exact schemes the operation count for multi-dimensional problems depends on the strategy adopted for the computation of the cross derivatives. Indeed, since the Schwartz equality does not hold exactly for the numerical solution, an average of different permutations of the derivation variables may be introduced to avoid dissymmetries, especially when irregular grids are used. In this case the operation count corresponds the upper bound, the lower one being obtained when only one permutation is computed. For 1D problems, increasing the order of accuracy from 2 to 3 doubles the operation count, and a 50% overcost is required to increase the order from 3 to 4. This is larger than the expected gain in terms of number of points per wavelength. For multidimensional problems, the overcost is increased due to the larger number of high-order derivatives required by the correction procedure. This is once again larger than the expected gain in terms of number of mesh points. Additionally, the memory requirements also grow significantly (e.g., the 3-exact scheme involves the storage of at least 10 third derivatives, if the Schwartz inequality is satisfied exactly).

Based on the preceding considerations, in the following we focus on k -exact schemes with orders of accuracy up to 3, which represent a reasonable compromise between dissipation and dispersion error requirements on the one hand, and computational complexity and memory occupation on the other. The third-order scheme seems particularly promising because it significantly reduces dispersion errors with respect to the second-order one, for a moderate overcost. Furthermore, it is known that concerning the dissipation errors, in the next Section we introduce a strategy allowing to improve the scheme resolvability in regions dominated by small vortical structures, which are susceptible to be overdamped, by suitably tuning the numerical dissipation terms in the scheme. We show that this strategy is more efficient than a simple increase of the order of accuracy.

Providing a general analysis of numerical errors on general unstructured multi-dimensional grids using purely analytical tools is not feasible. The proposed 1D analysis (on regular Cartesian grids) provides however a guide about the numerical scheme behavior in idealized conditions. A numerical methodology for analyzing dissipation and dispersion errors of spatial discretization schemes on general grids has been recently proposed by some of the present authors [44], based on the work of Takacs [45], and is going to make the object of a separated paper.

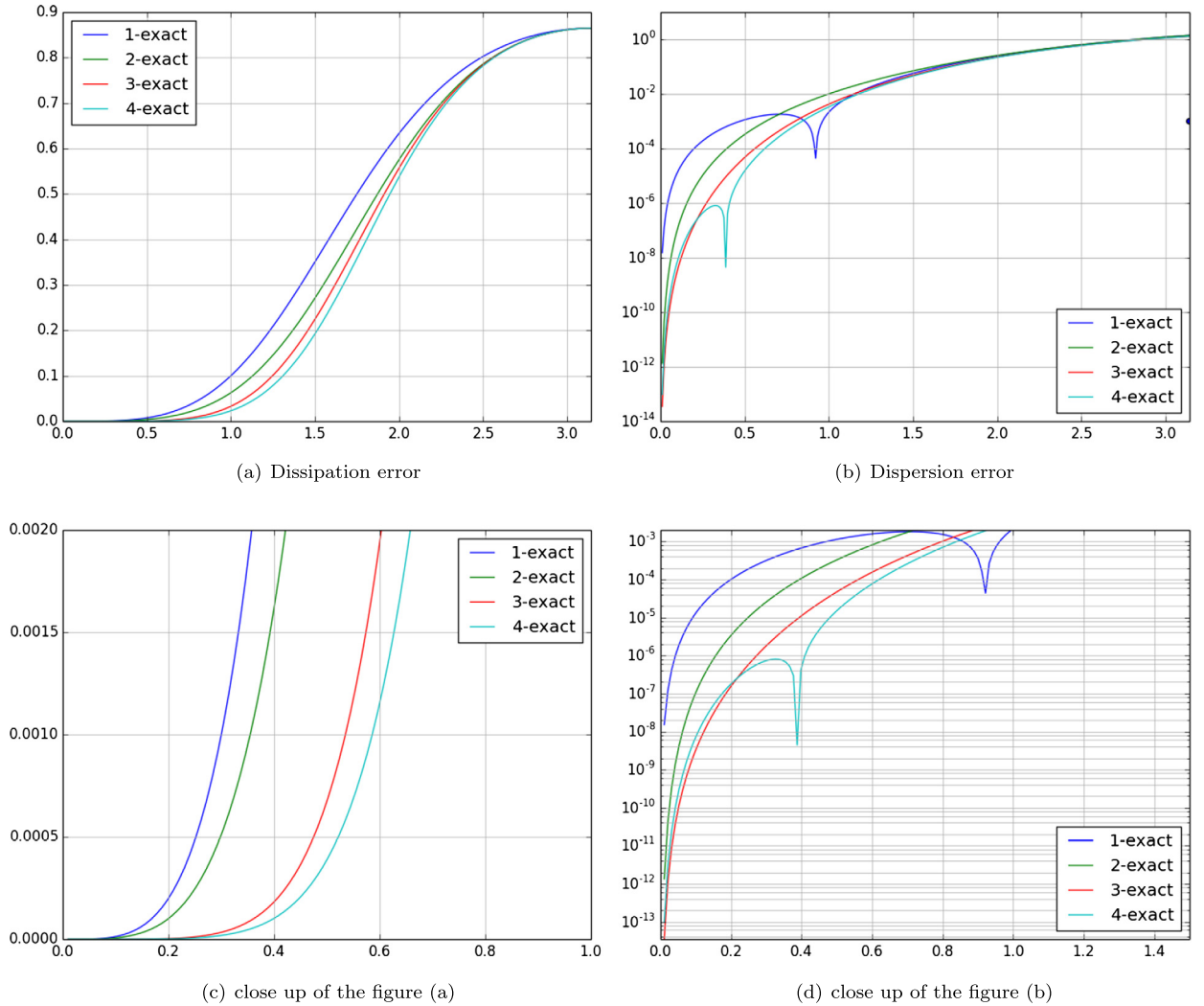


Fig. 2. Dissipation (\mathcal{A}) and dispersion (\mathcal{P}) errors for k -exact schemes of different orders.

5. Local centering of numerical fluxes in vortex-dominated regions

An important source of errors in RANS/LES simulations is due to by the numerical dissipation introduced by the approximation scheme in use for the inviscid fluxes (see e.g. [34,35,46]), which has to be as small as possible in order to preserve vortical structures. In the preceding Section, we introduced several upwind k -exact schemes. It was shown that the 3rd-order scheme, which represents a reasonable tradeoff between accuracy and computational cost, requires about 17 points per wavelength to achieve a sufficiently small damping error. This leads to unacceptably dense computational grids for practical applications. To reduce the numerical dissipation introduced by the upwind scheme in vortex-dominated regions, we adopt a local centering strategy, described hereafter. We aim at building a hybrid scheme, centered in vortical regions and upwind elsewhere. The resulting hybrid scheme is referred to hereafter as VC-scheme, standing for Vortex-Centered scheme. Similar approaches were developed in [34], where a hybrid scheme is constructed by blending of a fourth-order centered scheme in LES regions and of a third-order or fifth-order upwind scheme in RANS regions, and [35], where a local upwind correction is added to a purely centered, nominally second-order scheme wherever spurious grid-to-grid oscillations, or “wiggles”, are detected by means of a suitable sensor. Differently from Refs. [34,35], the present strategy automatically leads to a stable discretization, without need for empirical blending functions or oscillation sensors.

We start with rewriting the upwind spatial operator \mathcal{L} in Eq. (65) as the sum of a purely centered operator \mathcal{L}_C , and a numerical dissipation term \mathcal{L}_D :

$$\mathcal{L} = \mathcal{L}_C + \mathcal{L}_D \quad (79)$$

where:

$$\mathcal{L}_C = \frac{1}{2} (\mathcal{L}_R + \mathcal{L}_L),$$

with \mathcal{L}_L and \mathcal{L}_R the upwind and the downwind operators, respectively, and

$$\mathcal{L}_D = -\frac{1}{2} \text{sgn}(a) (\mathcal{L}_R - \mathcal{L}_L).$$

Introducing a blending function $\psi \in [0, 1]$ that multiplies the dissipation term, leads to the following hybrid scheme:

$$\mathcal{L}_{VC} = \mathcal{L}_C + \psi \mathcal{L}_D \quad (80)$$

where the suffix *VC* stands for “vortex centered” or, equivalently:

$$\mathcal{L}_{VC} = \frac{1 - \text{sgn}(a)\psi}{2} \mathcal{L}_R + \frac{1 + \text{sgn}(a)\psi}{2} \mathcal{L}_L, \quad \text{with } \psi \in [0, 1] \quad (81)$$

Clearly, for $\psi = 0$ the preceding scheme is fully centered and for $\psi = 1$ the upwind scheme is recovered.

For advection-diffusion problems of the form:

$$\frac{\partial \alpha}{\partial t} + a \frac{\partial \alpha}{\partial x} = (v + \nu_t) \frac{\partial^2 \alpha}{\partial x^2}, \quad (82)$$

it is well-known that a purely centered second-order accurate approximation of the convective terms leads to stable solutions under the standard stability conditions:

$$Re_g \leq 2; \quad CFL \leq 1$$

where $Re_g = \frac{|a|h}{v + \nu_t}$ is called the grid Reynolds number, based on the characteristic cell size h , and the sum of the molecular ν and eddy viscosity ν_t , and CFL is the well-known Courant–Friedrichs–Lewy number. A similar analysis can be applied, e.g., to a hybrid scheme of the form (80) with \mathcal{L}_C the purely centered second-order operator:

$$\mathcal{L}_C = a \frac{\alpha_{j+1} - \alpha_{j-1}}{2h}$$

and \mathcal{L}_D the numerical dissipation of the first-order upwind scheme:

$$\mathcal{L}_D = -\frac{|a|}{2h} (\alpha_{j+1} - 2\alpha_j + \alpha_{j-1})$$

With this choice, Eq. (80) reduces to the second-order centered scheme when $\psi = 0$ and to the first-order upwind scheme for $\psi = 1$. Using a forward Euler scheme for the time discretization, the corresponding difference equation writes:

$$\frac{\alpha_j^{n+1} - \alpha_j^n}{\Delta t} + a \left(\frac{\alpha_{j+1}^n - \alpha_{j-1}^n}{2h} \right) - \psi \frac{|a|}{2h} (\alpha_{j+1}^n - 2\alpha_j^n + \alpha_{j-1}^n) = (v + \nu_t) \frac{\alpha_{j+1}^n - 2\alpha_j^n + \alpha_{j-1}^n}{h^2} \quad (83)$$

or also:

$$\alpha_j^{n+1} = A\alpha_{j+1}^n + B\alpha_j^n + C\alpha_{j-1}^n \quad (84)$$

where:

$$A = -\frac{|a|(1 - \psi)\Delta t}{2h} + \frac{(v + \nu_t)\Delta t}{h^2} \quad (85)$$

$$B = 1 - \frac{|a|\psi\Delta t}{h} - \frac{2(v + \nu_t)\Delta t}{h^2} \quad (86)$$

$$C = \frac{|a|(1 + \psi)\Delta t}{2h} + \frac{(v + \nu_t)\Delta t}{h^2} \quad (87)$$

According to well known properties, the preceding difference equation preserves monotonicity, and therefore does not generate spurious oscillations, if all of the coefficients are positive. Note that when an implicit scheme of integration is used, the second term of B disappears and B is always positive. This condition is a CFL condition. C is always positive and only the $A \geq 0$ condition remains.

Clearly, from the third one of Eqs. (85) it appears that $C \geq 0$ for any positive h , Δt and ψ .

Table 3
Number of points per wavelengths as a function of the recentering function.

Re_g	∞	10	5	3.33	2.5	2.12	2.04	≤ 2
ψ	1	0.8	0.6	0.4	0.2	0.06	0.02	0
% recentering	0	20	40	60	80	94	98	100
Points per wavelength	17	14.1	12.3	11.7	10.4	8.8	6.73	no damping

The condition $A \geq 0$ is satisfied if:

$$\psi \geq 1 - 2 \frac{(\nu + \nu_t)}{|a|h} = 1 - \frac{2}{Re_g} \quad (88)$$

which leads to:

$$\psi \geq \max\left(0, 1 - \frac{2}{Re_g}\right), \text{ since } \psi \in [0, 1]. \quad (89)$$

For $\psi = 0$, the preceding expression reduces to the standard constraint on the grid Reynolds number for the purely centered scheme, whereas for a purely upwind scheme, $\psi = 1$ and the condition is satisfied for any Re_g .

Finally, $B \geq 0$ if:

$$\Delta t \leq \left(\frac{\psi}{\Delta\tau_1} + \frac{1}{\Delta\tau_2}\right)^{-1} \quad (90)$$

with $\Delta\tau_1 = h/|a|$ and $\Delta\tau_2 = h^2/2(\nu + \nu_t)$ are respectively the inviscid and viscous characteristic times.

When $\psi \geq \max\left(0, 1 - \frac{2}{Re_g}\right)$ (when the scheme is re-centered) we get:

$$\Delta t \leq \min(\Delta\tau_1, \Delta\tau_2), \text{ since } \psi \in [0; 1] \quad (91)$$

if $\psi = 1$ (when the scheme is fully upwind) we have:

$$\Delta t \leq \left(\frac{1}{\Delta\tau_1} + \frac{1}{\Delta\tau_2}\right)^{-1} \leq \min(\Delta\tau_1, \Delta\tau_2) \quad (92)$$

Equation (92) shows that re-centering strategy allows the CFL stability condition to be less restrictive.

In the following, we focus on the hybrid scheme obtained by applying the centering procedure to the 3rd-order k -exact upwind scheme. Such a scheme is somewhere in between the baseline scheme and a 4th-order, zero-dissipative scheme. Its truncation error is given by:

$$\mathcal{E}_{\mathcal{L}_{FV}^C}^3 = \psi \frac{h^3}{16} \frac{\partial^4 \alpha}{\partial x^4} \Big|_i - \frac{199h^4}{5760} \frac{\partial^5 \alpha}{\partial x^5} \Big|_i + \mathcal{O}(h^4)$$

This is similar to Eq. (74) with the dissipative error term multiplied by ψ . When $\psi = 0$ the dissipation term vanishes and the leading error term is 4th-order and purely dispersive.

The recentering procedure improves resolvability with respect to the baseline scheme: Table 3 gives the number of points per wavelength required by the hybrid scheme to achieve a dissipation error lower than 10^{-3} as a function of the centering parameter ψ and Fig. 3 shows the dissipation error for various value this centering parameter ψ . The number of required points per wavelength decreases when ψ is reduced from 1 to 0, the damping error being strictly zero when $\psi = 0$. For instance, the 80% recentered scheme requires a similar number of points per wavelength as the 4th-order scheme. When $\psi = 0$, the scheme is zero-dissipative and the necessary energy drain is ensured by only laminar and eddy viscosities. Finally, the introduction of the recentering function has an almost negligible impact on the computational cost of the simulation.

To investigate the stability of the k -exact hybrid scheme for advection-diffusion problems of the form (82), we further assume that viscous terms are approximated by a standard second-order scheme. For all of the computations presented in the following, time integration is performed via the second-order accurate Heun scheme, which writes for the present linear model problem:

$$\alpha^{n+1} = \alpha^n \left(1 + \mathcal{C}\Delta t + \frac{\Delta t^2}{2} \mathcal{C}^2 \alpha\right).$$

The stability region of the Heun scheme is given by the set of complex numbers $\widehat{\mathcal{C}}\Delta t$, such that:

$$\left|1 + \widehat{\mathcal{C}}\Delta t + \frac{(\widehat{\mathcal{C}}\Delta t)^2}{2}\right| \leq 1$$

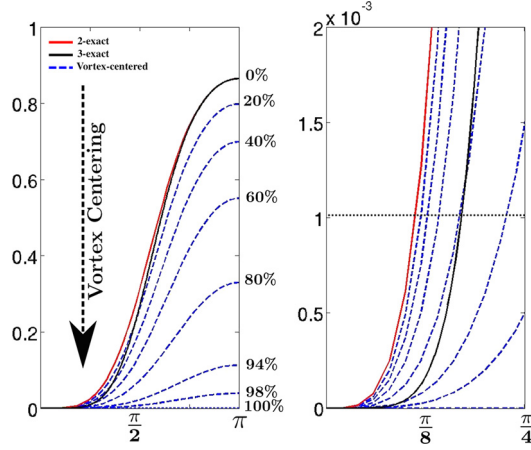


Fig. 3. Left: dissipation error of the hybrid scheme as a function of the centering parameter ψ . Right: close-up near the origin.

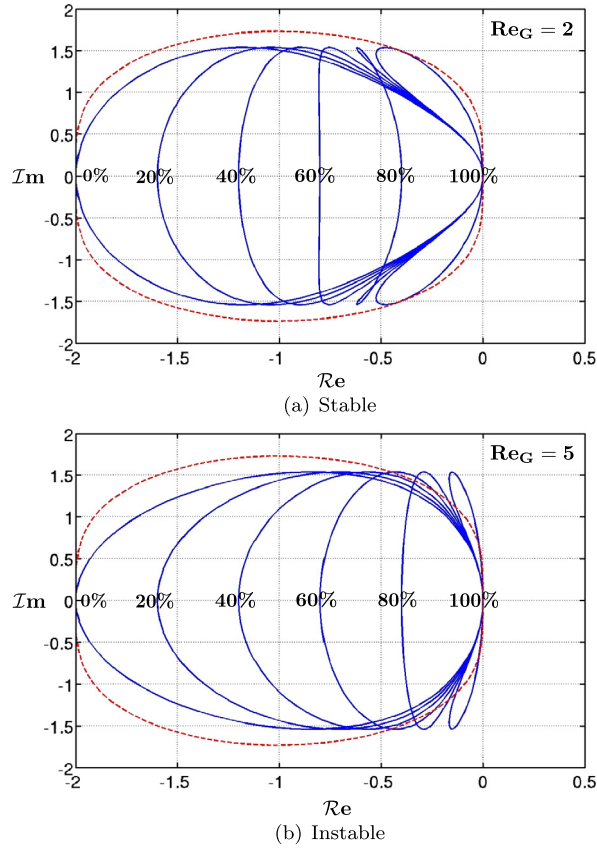


Fig. 4. Stability of the hybrid scheme coupled with a second order Heun temporal scheme. The red dashed curve represents the stability domain of the time operator, the blue curves the locus of the spatial operator for different values of the centering parameter. (For interpretation of the colors in this figure, the reader is referred to the web version of this article.)

where \hat{C} represents the Fourier symbol of the spatial discretization. The boundary of this region is represented by the red curve in Fig. 4. For the hybrid spatial discretization scheme under investigation, the Fourier symbol \hat{C} depends on the reduced wave number under consideration and on ψ , the CFL number and Re_g . The fully discrete scheme is stable when the locus of \hat{C} as a function of the reduced wave number must lie within the stability region of the time discretization scheme. Fig. 4 shows the locus of the spatial approximation for different choices of ψ and two different choices of the grid Reynolds number, namely, $Re_g = 2$, which corresponds to the upper acceptable value for the purely centered second-order scheme, and $Re_g = 5$. The CFL number is taken equal to 1. For lower values of CFL, the locus of the spatial operator is smaller and

the resulting fully discrete scheme is certainly stable. When the scheme is fully upwind, i.e. for $\psi = 1$ (0% recentering) the scheme is stable for any Re_g . When the scheme is 100% centered ($\psi = 0$), then the spatial locus is tangent to the boundary of the stability domain when $Re_g = 2$, which corresponds to the limit value according to Eq. (88), and partly outside of it for $Re_g > 2$. Also note that it is possible to obtain a stable scheme for $Re_g > 2$ by using just a partial recentering. For instance, for $Re_g = 5$, stability is achieved if the scheme is recentered by about 40% or less, i.e. for $\psi \gtrsim 0.6$, which is in agreement with the simple relation (88).

In practice, ψ is not constant throughout the computational domain, but is allowed to vary between 0 and 1 according to the local flow features: in regions dominated by fine vortical structures ψ has to be as close to zero as possible, whereas in regions characterized by strong compressibility effects, like shock waves, ψ must tend to 1 to ensure robustness. In this work, the blending function is related to the vortex sensor introduced by Ducros et al. [47], defined as:

$$\Phi = \frac{(\text{div}(\mathbf{u}))^2}{(\text{div}(\mathbf{u}))^2 + |\text{rot}(\mathbf{u})|^2}. \quad (93)$$

This sensor tends to 0 in nearly incompressible and/or highly rotational regions, whereas it tends to 1 in highly compressible and/or irrotational regions. Moreover, according to the preceding stability analysis, in regions where $\psi = 0$, 100% centering is allowed only if $Re_g \leq 2$, otherwise the local recentering rate has to satisfy the constraint (88). This is achieved by defining the local blending function as follows:

$$\psi = \max\left(\Phi, 1 - \frac{2}{Re_g}\right) \quad (94)$$

With the preceding formulation, only partial recentering is allowed in vortex-dominated regions with insufficient grid resolution, which ensures the minimal amount of numerical dissipation to damp grid-to-grid oscillations.

The computational overcost introduced by the local centering procedure is practically negligible compared to the cost of the baseline scheme. It is mainly due to the calculation of the discrete counterpart of Ducros' sensor, which involves an approximation of the divergence and of the rotational of the velocity field, and the grid Reynolds number. The required derivatives are already available in the solver, since they are needed for the polynomial reconstruction of the solution. Furthermore, at cell interfaces both the upwind and the downwind numerical fluxes can be computed in a straightforward manner from the reconstruction operated on the left and right cells. Thus, the recentering procedure is a simple blending of already available terms and does not introduce a significant number of additional operations.

6. Numerical results

In this Section, the second and 3rd-order k -exact upwind schemes and the hybrid VC scheme are first applied to some well documented test cases. The objective is twofold: i) to demonstrate the capability of k -exact schemes to ensure high-order accuracy on general grids; ii) to prove the effectiveness of the VC strategy to preserve vortical structures while preventing spurious oscillations.

6.1. Preliminary validations: Ringleb flow

To verify the accuracy of the preceding numerical schemes, we consider an inviscid transonic test case, the so-called Ringleb flow [48], for which an exact solution for the steady aerodynamic field is available. Details about this configuration can be found in [48]. The analytical solution is defined implicitly along streamlines, and leads to the inversion of a system. Such solution is then integrated over each cell by means of 6th-order Gauss quadrature formula, to obtain initial values for the cell-averaged conservative variables. These are given as an input to the finite-volume solver. Using a 6th-order quadrature formula avoids polluting the initial solution with integration errors that would be comparable to the order of accuracy of the scheme in use. The exact solution is also used to enforce the boundary conditions. Initially, calculations are run using the second and third-order accurate upwind schemes on a set of smooth computational grids composed by 192, 768, 3072, 12288 and 49152 quadrangular cells, respectively (Fig. 5 (b)). Afterwards, the capability of the scheme to preserve accuracy on highly deformed grids is demonstrated by using a set of grids obtained by randomly shaking the nodes of the preceding ones (Fig. 5 (c)). Finally, we perform calculation on hybrid grids involving both quadrangular and triangular cells (Fig. 5 (d)), stretched quadrangular grids (Fig. 5 (e)) and stretched triangular grids (Fig. 5 (f)). The height of the first cell close to the side boundaries of the stretched grids is equal to 0.01 m. Computations are carried out using a CFL of 0.9 and are stopped when the non-dimensional residual reach 10^{-7} .

For each grid, the L_2 norm of the error with respect to the exact solution for the cell-averaged primitive variables resulting from the k -exact reconstruction is plotted versus a typical mesh size $h = 1/\sqrt{DOF}$, with DOF the number of degrees of freedom, here equal to the number of cells. The results are represented in Fig. 6 for the longitudinal velocity. Similar results are obtained for the other primitive variables. The computed convergence orders are in fair agreement with the nominal ones, both for the second-order and the third-order upwind scheme. Using highly irregular grids does not affect the convergence order. For grids containing triangular cells, error slopes in agreement with second- and third-order accuracy are recovered for the 1- and 2-exact schemes, respectively, although with somewhat higher errors levels than the quadrangular grid. The overall flow field is anyway little affected (see solution isocontours reported in Fig. 7).

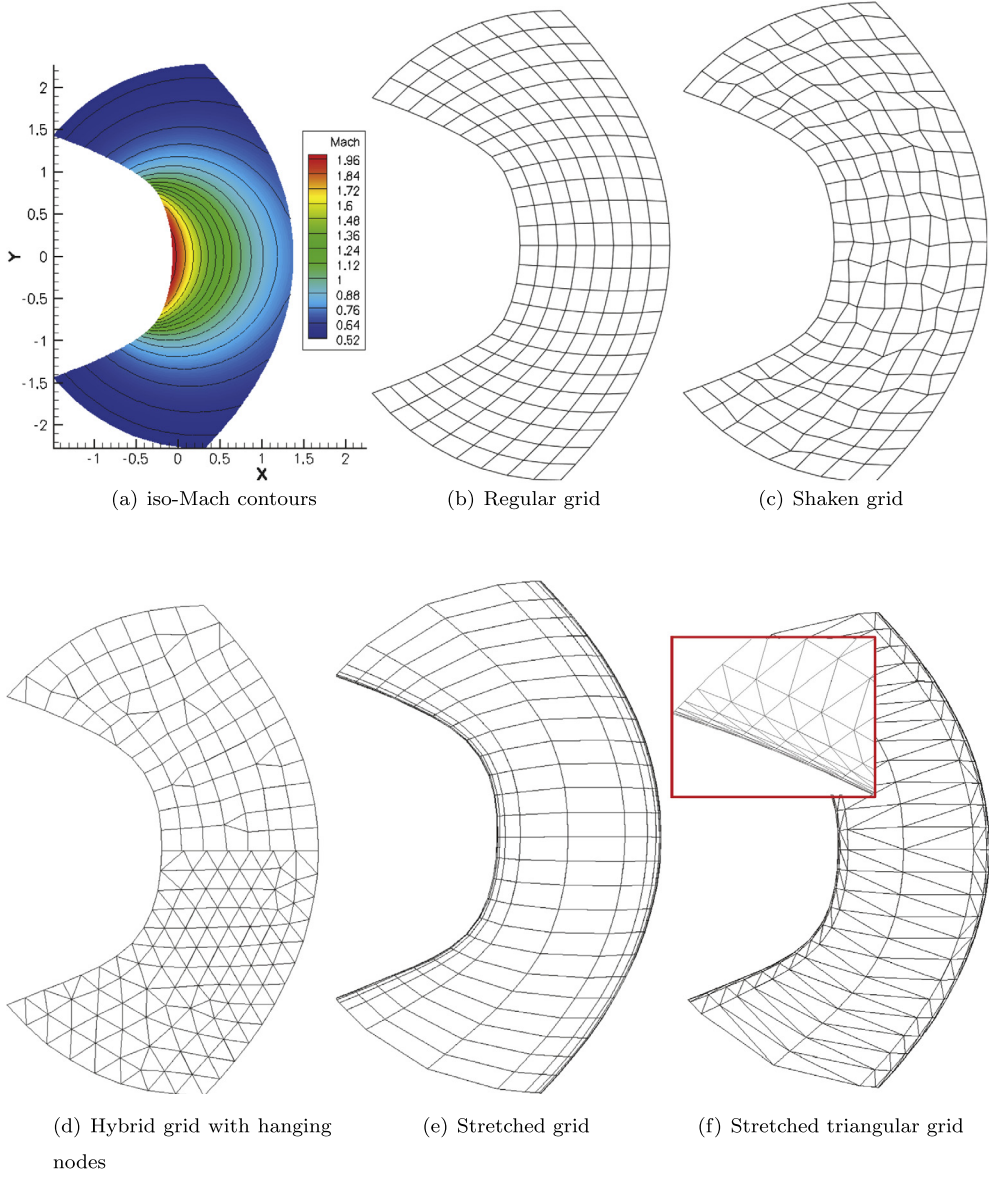


Fig. 5. Ringleb flow: exact solution and types of grids used for the calculations.

6.2. Inviscid advection of a Lamb vortex

In the following we consider the advection of a Lamb vortex [49] by an inviscid uniform flow with Mach number equal to $\sqrt{2}$. In a cylindrical reference frame attached to the vortex center, the azimuthal velocity, temperature and pressure field are prescribed through the following relations:

$$V_{\theta}(r) = \frac{\Gamma}{2\Pi r} \left[1 - \exp(-\eta^2) \right] \quad (95)$$

$$T(r) = \frac{\Gamma}{\Pi a_0^2} \left[-\exp(-\eta^2) \right] \quad (96)$$

$$P(r) = \frac{\Gamma^2}{8\Pi^2 a_0^2} \left[\frac{1}{\eta} \left(1 - \exp(-\eta^2) \right)^2 + g(\eta) + g(2\eta) \right] \quad (97)$$

where $\eta = r/a_0$, $g(\eta) = 2 \int_{\eta}^{\infty} \exp(-\xi) d\xi$. The solution is computed on a set of 7 square grids with a number of equally spaced quadrangular cells ranging from 40000 to 640000. A very small time step is used in order to rule out numerical errors related to the time integration method. Periodic boundary conditions are imposed at grid borders. The vortex is

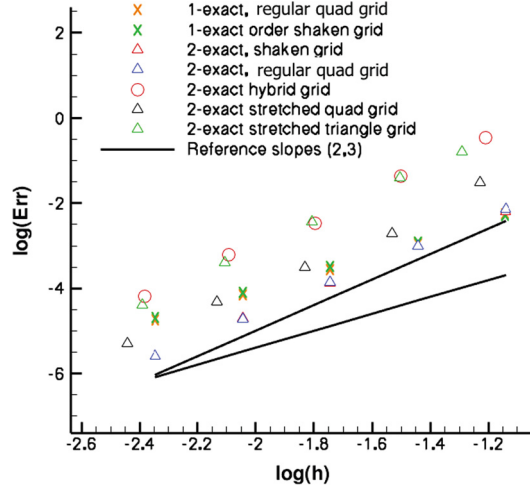


Fig. 6. Ringleb flow. Grid convergence of the pressure field. $h = 1/\sqrt{nDOFs}$, where nDOFs is the number of degrees of freedom.

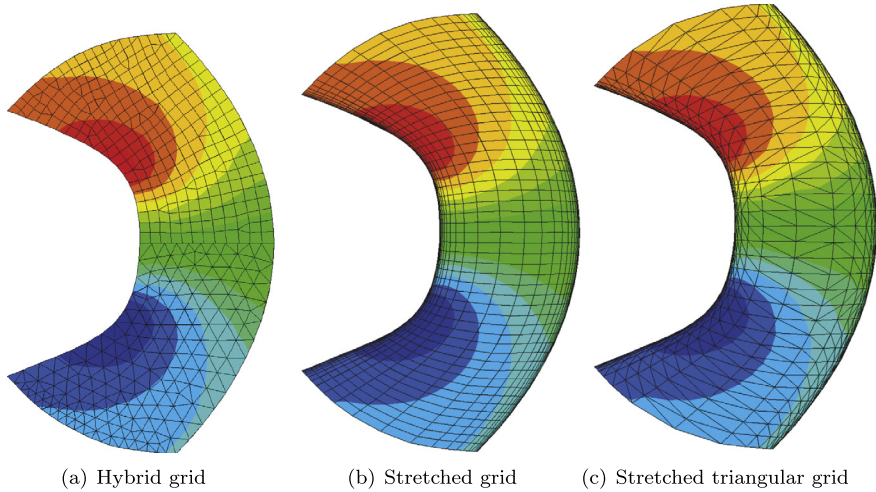


Fig. 7. Isocontours of the x velocity component on various grids.

advected along the mesh diagonal over a length of three diagonals. To demonstrate the ability of the present method to preserve high accuracy on any grid, simulations are also run on a set of irregular quadrangular grids, obtained by randomly shaking the nodes of the preceding regular grids. Fig. 8 shows a close-up of a Cartesian and a shaken grid. Similarly to the preceding test case, we investigate the grid convergence of the numerical solution and, specifically, of the conservative variables. Fig. 9 shows the L_2 norm of the error with respect to the exact solution for the pressure field, versus the characteristic mesh size h . Similar results are obtained for the other variables. The computed convergence orders obtained for the 2nd-order and the 3rd-order k -exact schemes are in fair agreement with the nominal ones, both on Cartesian and shaken grids. This is not achieved when using, for instance, a standard upwind finite volume scheme using a 1D 3rd-order reconstruction of the variables along mesh lines, without corrections for mesh deformations. For this scheme, the convergence order is equal to 1 on the highly deformed shaken grids.

For a more detailed analysis of the k -exact approach, the convergence order of the 1st and 2nd derivatives of the computed field is also investigated. For this purpose, different numerical approximations of the derivatives are compared to the analytical derivatives available for the exact solution.

Fig. 10 shows the convergence of the L_2 error of the pressure gradient versus the mesh size for a standard approximation based on the standard Gauss–Green relation (36), (37) – used in most finite volume solvers –, for the consistent 1-exact first differentiation operator (44), obtained by correcting (36), (37) by the linear operator \mathbf{M}_1 , and for the 2-exact first differentiation operator resulting from the successive correction procedure of Section 3.4. The 1-exact and 2-exact operators provide, respectively, a 1st-order and a 2nd-order approximation of the gradient on any grid, whereas the Gauss–Green operator becomes inconsistent on irregular grids. In particular, the introduction of the correction operator \mathbf{M}_1 is what makes the difference between the 1-exact and the standard operator. This correction is then an essential ingredient to ensure a

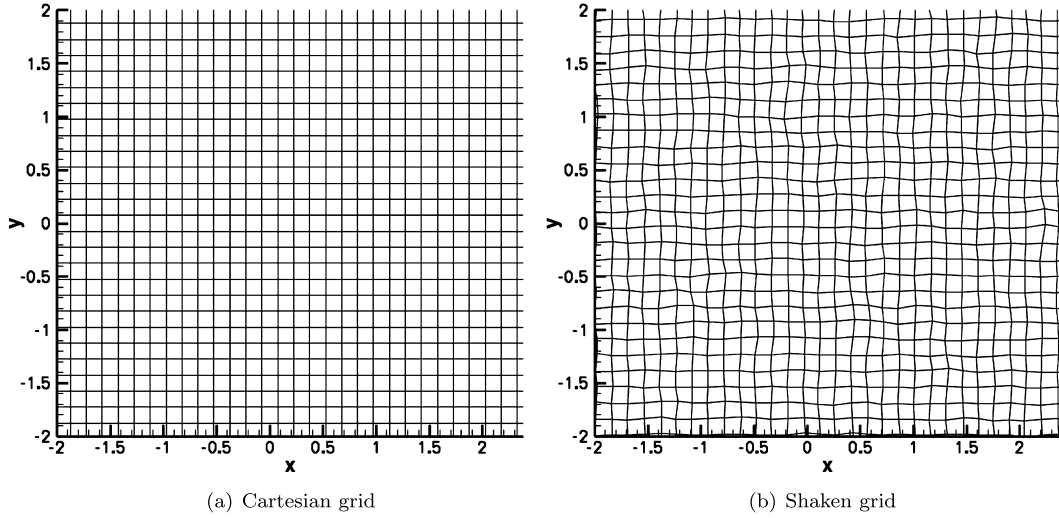


Fig. 8. Computational grids for the lamb vortex advection.

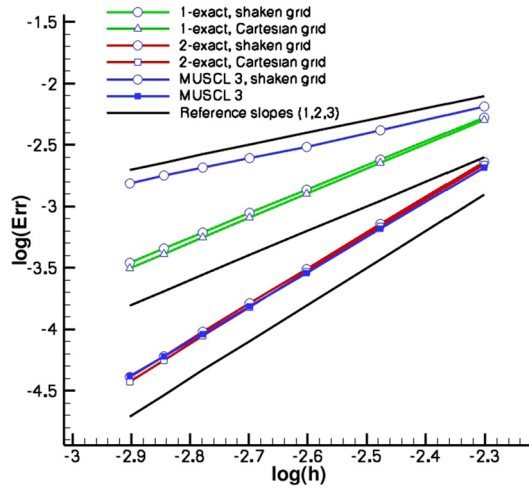


Fig. 9. Lamb vortex advection. Grid convergence of the L_2 norm of the error for the pressure field with different schemes.

consistent approximation for the gradients, independently on the grid in use. Similar results are obtained for the gradients of other variables, like velocity or temperature.

Similarly, Fig. 11 shows the convergence of the L_2 error for second derivatives of the pressure field. In this case, a baseline approximation obtained by applying twice the 1-exact first differentiation operator is compared to a first-order 2nd-differentiation operator obtained by correcting the preceding one by means of the linear application \mathbf{M}_2 (see Eq. (58)). Only the first-order accurate operator ensures 1st-order convergence of the second-derivatives on any grid, whereas the straightforward procedure leads to an inconsistent solution.

Note that flow solvers often require approximations of the first and second derivatives, for instance to compute source terms in transport-equation turbulence models. Using standard Gauss–Green approximations of the gradients is inconsistent, especially on highly deformed grids often used to simulate industrial configurations, leading to significant numerical errors.

Finally, it is interesting to quantify the influence of the $\Delta\mathbf{q}$ correction on the grid convergence of the method. On the shaken grid, a convergence order equal to the nominal one is recovered when applying the correction $\Delta\mathbf{q}$, otherwise the slope is reduced to 2.5 (see Fig. 12).

6.3. Viscous vortex advection and shock-vortex interaction

In addition to k -exact reconstructions, another important ingredient of the present numerical strategy is the vortex-centering procedure illustrated in Section 5. This problem, initially proposed by Shu [50] is usually computed under an inviscid flow hypothesis. Here, we consider a viscous flow in order to investigate the effect of the VC procedure. To validate the VC procedure illustrated in Section 5, we consider just only the vortex advection but we include viscous effects, so that

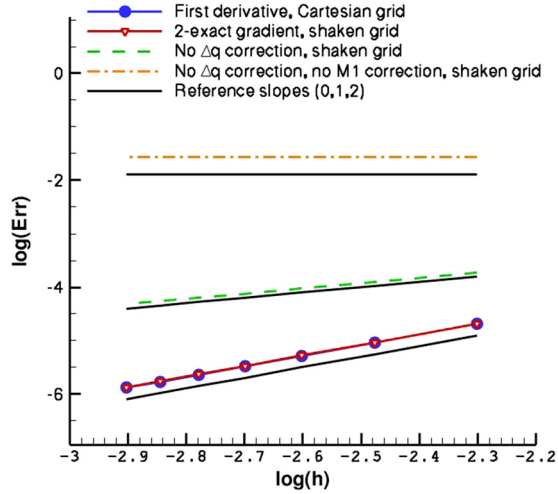


Fig. 10. Lamb vortex advection. Grid convergence of the L_2 norm of the error for the pressure gradient computed by the full 2-exact operator, the same applied directly to $\tilde{\mathbf{q}}$ with no $\Delta \mathbf{q}$ correction, and the operator without the M_1 correction for grid deformations. The latter is second-order accurate on the Cartesian grid and non consistent on the shaken one.

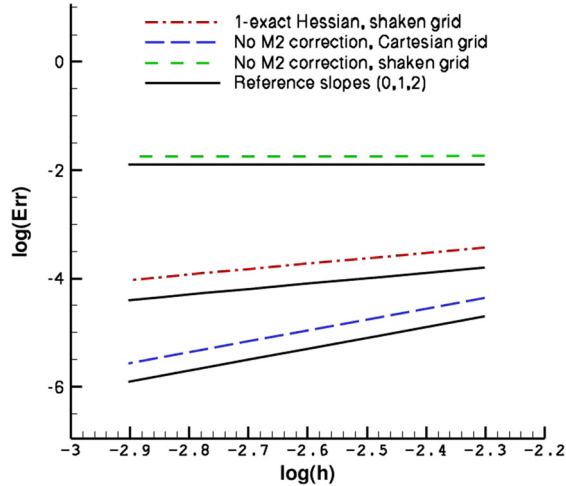


Fig. 11. Lamb vortex advection. Grid convergence of the second derivatives of the pressure field, computed by the 1-exact operator, and the same without the M_2 correction for grid deformations. The latter is second-order accurate on the Cartesian grid and non consistent on the shaken one.

it makes sense to define a grid Reynolds number. The vortex is initially located at point $X = Y = 0$ in the computational domain, and it is advected by an uniform flow with Mach number equal to 0.3. Simulations are carried out on three grids of increasing density, composed of 35×250 , 50×375 and 120×860 cells, respectively. Time integration is stopped after an elapsed time corresponding to 50 core length. The time step in use corresponds to a CFL of about 0.95. A reference solution is obtained by running the code on a very fine mesh (100 grid points in the vortex core). Numerical results displayed in Fig. 13 are in agreement with the truncation error study (equation (76)): the 2nd-order k -exact scheme is mainly dispersive and it introduces a phase lead of the numerical solution with respect to the physical one, in addition to significant numerical dissipation. The third-order scheme is less dispersive (its dispersion error is fourth-order and is based on fifth derivatives) and predicts a final vortex location that is close to the exact one, but it is yet too dissipative. The local centering procedure has no impact on dispersion errors, but it reduces numerical dissipation significantly, both for the second-order and the third-order scheme.

In the following, only the third-order upwind and the VC scheme are retained for further investigation.

Then, the fully upwind 2-exact and the vortex-centered scheme are applied to the computation of the vortex interaction with a steady shock with an upstream Mach number 1.1. The computational domain is the same we used in the previous test cast, we just add a steady shock in the middle of the computational domain. Computations are carried out on a grid composed by 50×375 quadrangular cells, such that the vortex core is discretized by 10 points. Fig. 14 presents a close-up of the vorticity contours and the pressure isolines computed by using the two schemes, at the middle of the computational domain, when the vortex crosses the shock. Both schemes capture the shock in a sharp and non-oscillatory way. The weak

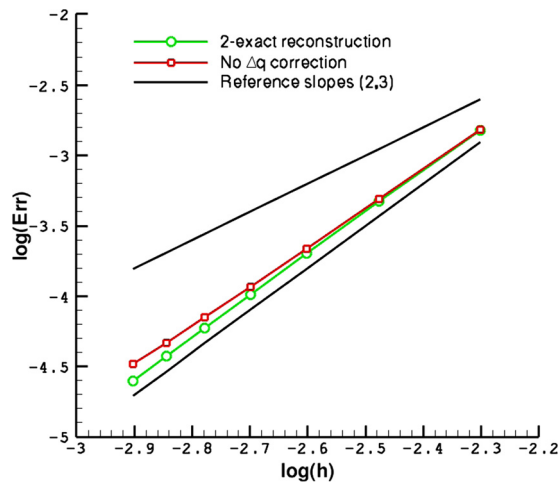


Fig. 12. Lamb vortex advection. Influence of the Δq correction on the convergence order of the solution (pressure field) for a shaken grid.

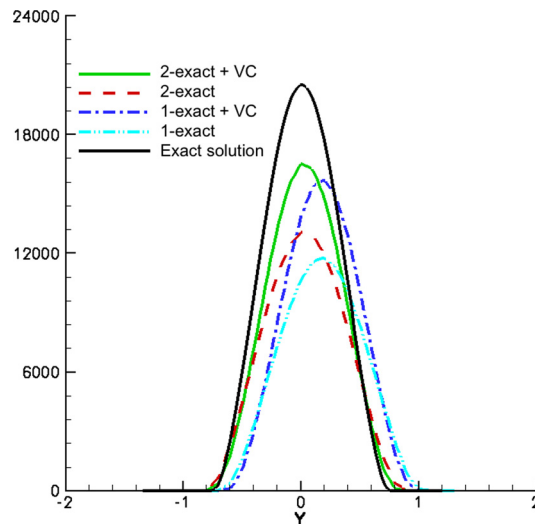


Fig. 13. Viscous vortex advection. Distribution of the Q-criterion across the vortex for fully upwind and vortex-centered schemes, after a traveled distance equal to 50 times the vortex core radius.

unsteady shocks generated by the interactions are also reasonably well captured by both schemes. The VC scheme better preserves vorticity than the upwind scheme, leading to a stronger interaction. This is better seen in Fig. 15, which shows the vorticity distribution along the line $y = 0$ for the two schemes under investigation. The numerical solutions are compared to a reference generated by using a grid 10 times finer in both directions. Finally, Fig. 16 shows the contours of the Ducros criterion. It can be seen that both vertical and shock regions are identified clearly, so that the scheme remains upwind across shocks and in irrotational regions and is centered elsewhere.

6.4. Viscous Taylor–Green vortex

The viscous Taylor–Green Vortex (TGV) is an unsteady viscous flow problem dominated by energy transfer from large to small flow scales. This problem has been proven to be an excellent testing bench to study the resolvability properties of high-order schemes in view of subsequent application to fine-scale turbulence simulations [51–53,46].

A three-dimensional vortex is set as an initial condition for 3D-computation. Because of vortex stretching and vortex tilting mechanisms, the vortex breaks up, giving origin to smaller and smaller structures. At finite Reynolds number, the kinetic energy is transferred from larger to smaller scales and dissipated by the smallest one; the test case gives thereby a simple model of the energy cascade. The Taylor–Green vortex is an example of nonlinear fluid flow with kinetic energy transfer from large to small scales and is a good milestone to assess the applicability of numerical schemes to Large Eddy Simulation (LES).

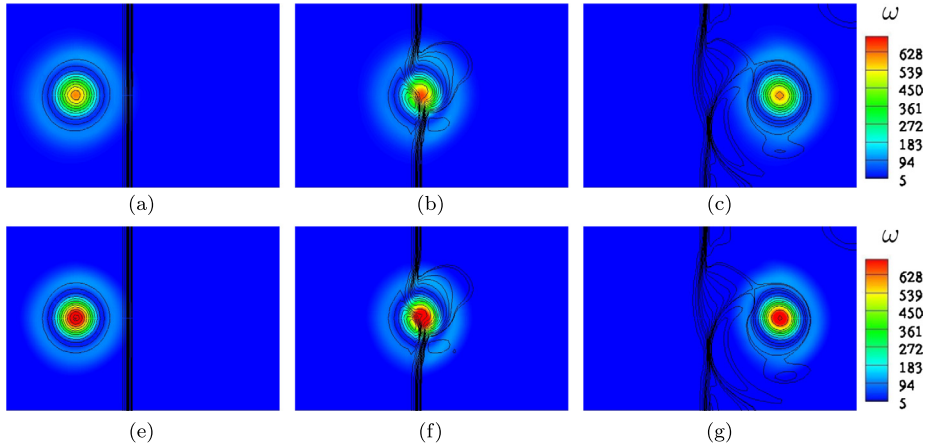


Fig. 14. Shock/vortex interaction. Vorticity contours and pressure isolines at times $t = 0.048$ (top), $t = 0.053$ (middle), and $t = 0.06$ (bottom). Left column: fully upwind 3rd order scheme; right column: VC scheme.

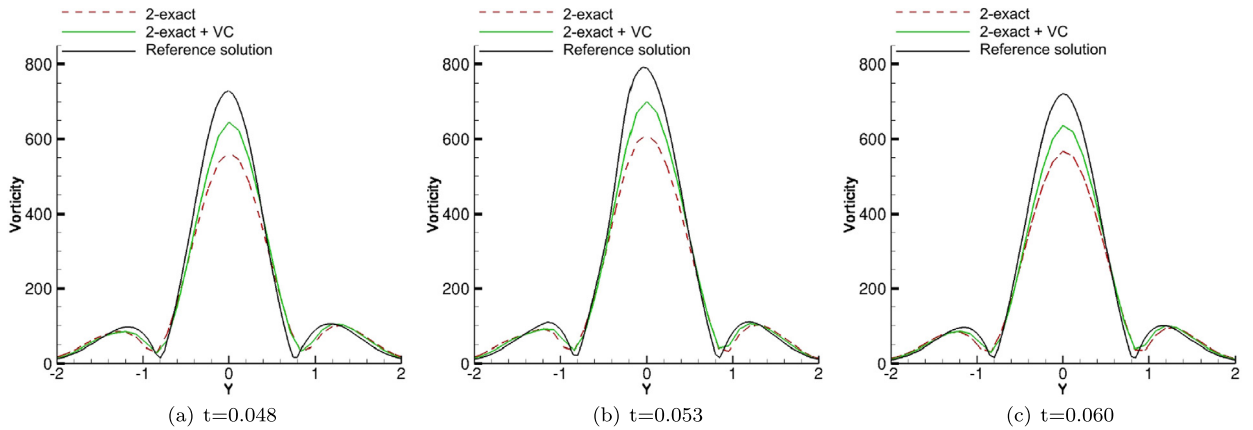


Fig. 15. Shock/vortex interaction. Vorticity distribution along the centerline $y = 0$.

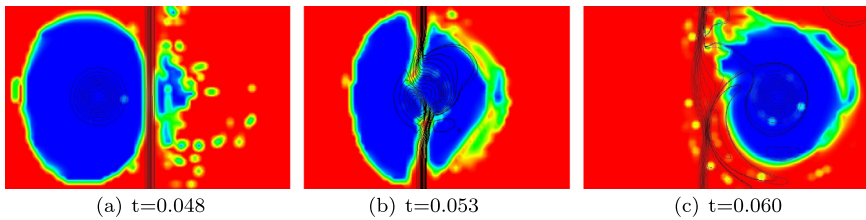


Fig. 16. Shock/vortex interaction. Contours of the Ducros sensor at different times.

Computations are run in a period cubic domain corresponding to $(0, 2\pi)^3$. The Reynolds number is set to 1600 and the average initial Mach number is taken small enough to reproduce an incompressible flow behavior. We run computations using the 2-exact upwind and the VC scheme using randomly shaken grids of different resolutions and precisely: 128^3 , 192^3 , 256^3 , 330^3 cells, and compare the results to a DNS obtained on a 512^3 with a pseudo-spectral, provided as a reference for the First International Workshop on High-Order CFD Methods held at the 50th AIAA Aerospace Meeting (<https://www.grc.nasa.gov/hiocfd/>).

Precisely, we consider the time evolution of the kinetic energy dissipation rate, i.e. the time derivative of the kinetic energy integrated over the computational domain, and of the global enstrophy, i.e. the integral of the local enstrophy per unit volume over the computational domain. The required derivatives and integrals are calculated by using approximation formulae of higher order than the numerical schemes under investigation, to avoid polluting the numerical solutions with post-processing errors.

Fig. 17 shows the effect of mesh resolution on the grid convergence of the time evolution of the enstrophy. The solution converges progressively toward the reference DNS both for the upwind and the VC scheme, even if the grid resolution

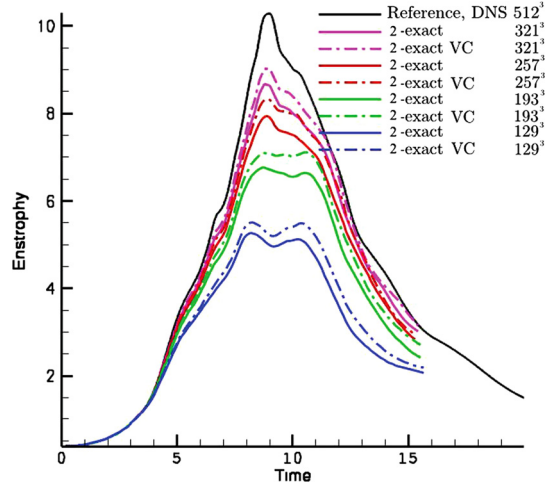


Fig. 17. Taylor–Green vortex at $Re = 1600$. Time evolution of the total enstrophy.

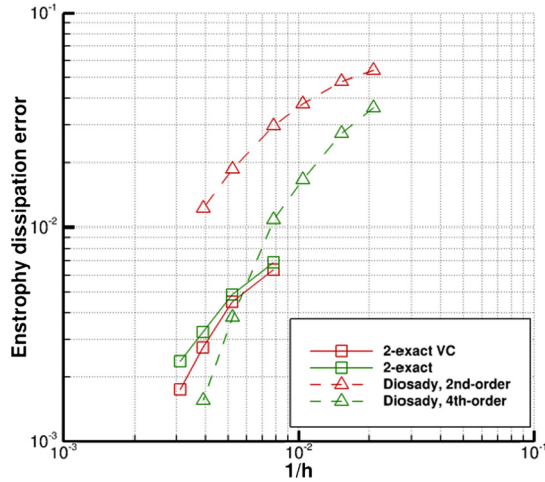


Fig. 18. Taylor–Green vortex at $Re = 1600$. Maximum errors on the enstrophy dissipation with respect to the reference DNS data, as a function of grid resolution $1/h$. This is defined as the number of degrees of freedom, DOF, to the power $-1/3$.

used in the present study is too coarse for DNS. The VC scheme preserves small structures created after the initial vortex-stretching phase better than the fully upwind scheme, leading to a higher enstrophy peak. Precisely, the solution of the VC scheme on the 256^3 grid is almost identical to the solution of the upwind scheme on a 320^3 grid, which implies a gain of more than a factor of 2 in terms of computational cost (the time step allowed on the coarse grid being 1.25 times greater than the one used on the finer one).

The present results were compared to selected results from the 3rd International Workshop on high-order methods. These were obtained by using second and fourth-order accurate Discontinuous Galerkin (DG) Methods [54]. Specifically in Fig. 18 we compare the errors in L_∞ norm on the enstrophy dissipation, i.e. the enstrophy multiplied by 2ν , with respect to the reference DNS. Results provided by present 2-exact scheme (with and without VC) compare rather favorably with the results provided by the fourth-order DG method, leading to approximately the same errors for comparable grid resolutions. The asymptotic slope of the VC scheme is, as expected, close to the one of the fourth-order scheme, whereas it is close to a third-order one for the baseline scheme.

6.5. Transonic turbulent flow past a square cylinder

In order to investigate the behavior of the VC sensor and to assess the capability of the proposed hybrid scheme to robustly capture unsteady shock waves while resolving the most relevant turbulent structures, we consider the transonic turbulent flow around a square cylinder at a Mach number of 0.9049 and a Reynolds number of 4.137×10^5 based on the cylinder side and free-stream conditions. The flow conditions are set as to mimic the experiment by Nakagawa [55]. The problem is quite challenging for numerical solvers, due to the presence of boundary layers, vortex shedding into the wake,

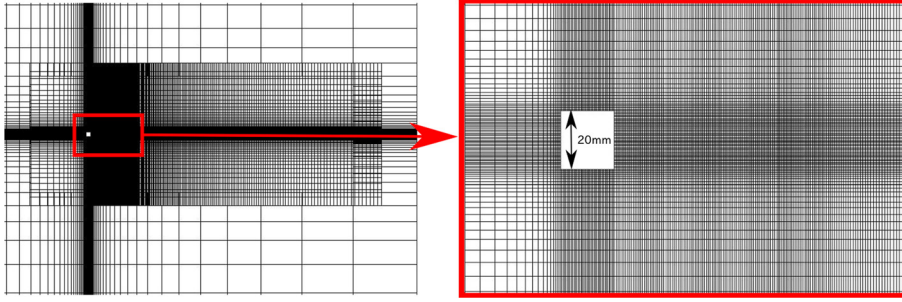


Fig. 19. Flow past a square cylinder. Views of patched grid.

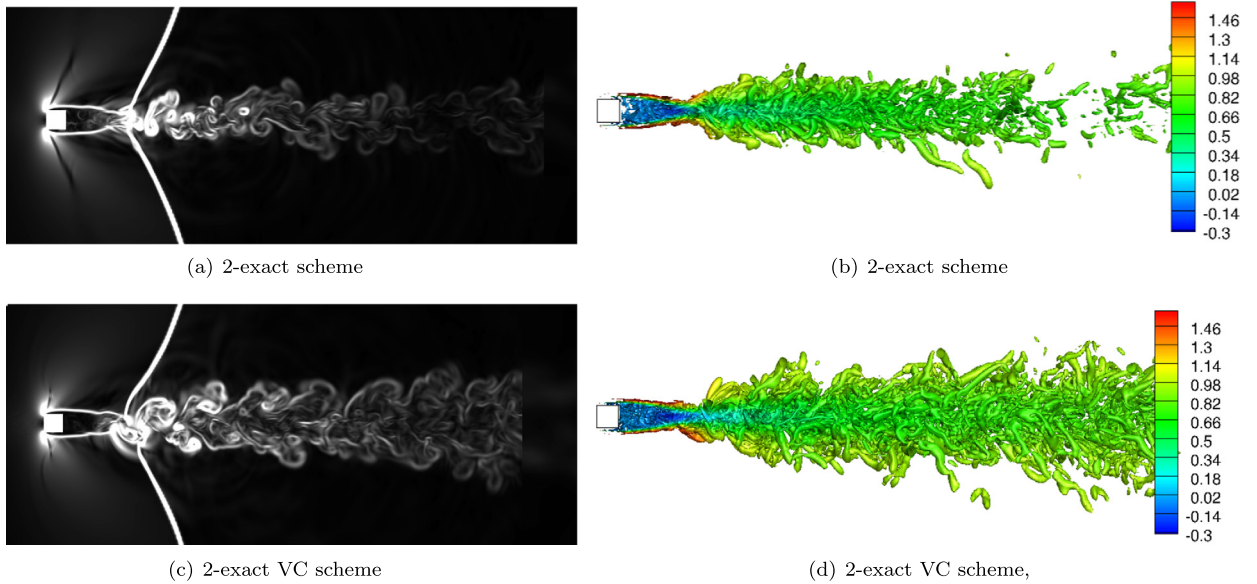


Fig. 20. Instantaneous numerical Schlieren in the middle plane and iso surface of $Q D^2 / U_0^2 = 0.1$ colored by mean normalized mean velocity: impact of the VC scheme.

interactions between Kelvin–Helmholtz instabilities in the shear layers and the von Karman vortex street, and shock waves. Turbulence effects are taken into account by using the DDES hybrid RANS/LES model. The simulations are carried out on a computational grid containing approximately 5 millions hexahedral cells with a y^+ of about 10 on the upper and lower surfaces of the cylinder. Wall functions are applied close to solid walls. An assessment of the HRL model is provided in Refs. [56,57]. Fig. 19 shows the patched grids used to discretize the computational domain. Standard wall functions are used to model the flow behavior in the viscous sublayer. The grid intersection strategy is described in [58–60]. The RANS/LES simulation is initialized by using a preliminary RANS simulation. The aim of this test case is to test the robustness of the flux-centering procedure and the effectiveness of the proposed vortex sensor.

Fig. 20 (b) and (d) provide show instantaneous numerical Schlieren (density gradient iso-contours), as well as an iso-surface of the Q -criterion colored by the normalized mean velocity. The results are obtained by using respectively the 2-exact scheme without and with the VC correction. The wake is characterized by a von Karman street that interacts with the unsteady shock waves generated by deflection of the locally supersonic flow downstream of the cylinder. The wake computed by using the VC scheme is characterized by finer turbulent structures, suggesting the numerical damping is effectively reduced by the VC procedure. We also note that the VC scheme captures the strong shocks accurately and robustly, without generating any spurious oscillations.

For a more quantitative validation of the proposed strategy, we investigate the dominating frequencies of the flow using sensors located behind square cylinder, in the middle plane along the horizontal axis of the computational domain. This are uniformly distributed over a distance $12D$ from the cylinder, D being the cylinder side. The normalized pressure spectra at location $x/D = 4$ is represented in Fig. 21 as a function of the Strouhal number (St), based on the free-stream conditions and cylinder side. The spectra exhibit peaks for St about 0.18, which is characteristic of von Karman vortex streets, and in a good agreement with the study of Nakagawa [55] (Strouhal number of 0.1817). The spectra exhibit also a peak corresponding approximately to a Strouhal number of about 0.5. This could be due to the vortices coming from the shear layers of the

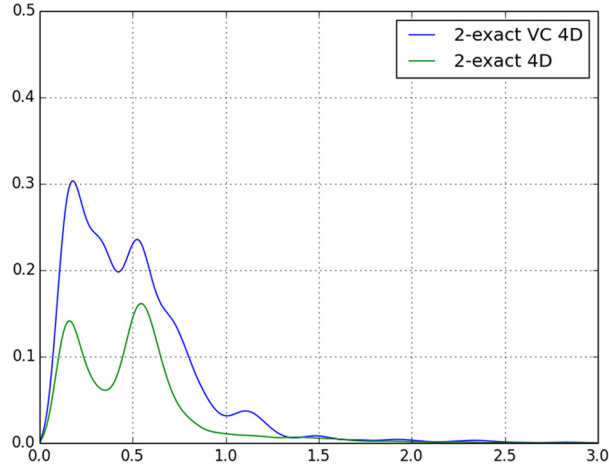


Fig. 21. Transonic flow past a square cylinder. Normalized Power Spectral Density of pressure ($fPSD/\sigma^2$) as a function of the Strouhal number (fD/U_{inf}) in the wake of the cylinder ($x/D = 4$).

Table 4
Grid parameters.

Mesh boxes	Number of cell	y^+	Type
Blue	~3 million	~100	Wind tunnel
Green	~10 million	~50	LES resolution
Red	~3 million	~25	Cylinder boundary layers

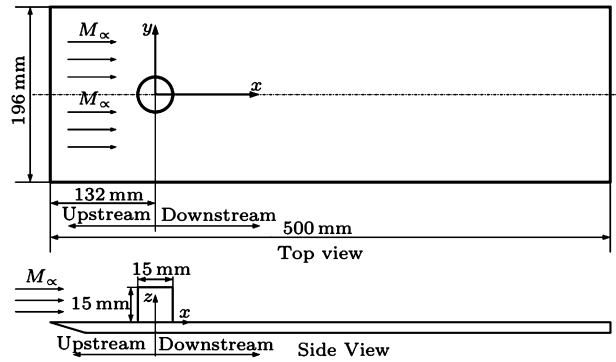


Fig. 22. Sketches of the computational setup.

square cylinder even if the grid used for this case is not fine enough to resolve correctly the Kelvin–Helmholtz instabilities in the shear layer observed for example in the experimental study of Williamson [61].

Most importantly, Fig. 21 shows that the energy content of spectra computed with the VC scheme are characterized by a higher energy content in the entire spectrum, compared to the baseline 2-exact scheme.

6.6. Supersonic flow over a finite cylinder on a flat plate

To demonstrate the applicability of the proposed numerical strategy to high Mach number HRL simulations we carry out numerical simulation of a supersonic flow around a finite circular cylinder mounted on a flat plate. This configuration has been studied experimentally by Wang et al. [62]. The flow field is characterized by complex three-dimensional shock waves and by a vortical wake. Sketches of the computational setup are provided in Fig. 22. The incoming flow Mach number is equal to 2.86 and the freestream pressure and temperature are 1 atm and 300 K, respectively. The grid used for the simulation is composed by three intersecting Cartesian patches (Fig. 23). Details of the grid are presented in Table 4. After intersection of the three patches, the resulting single-block unstructured grid is composed of about 12 million cells. Figs. 23 (c)–(e) are close up views of the grid about the cylinder, characterized by arbitrary quad elements. To simulate the wind tunnel walls, slip conditions are imposed at the top and side boundaries of the outer grid patch. Wall functions are applied at the flat plate wall in the RANS part of the calculation.

A DDES simulation is carried out using the vortex-centered scheme.

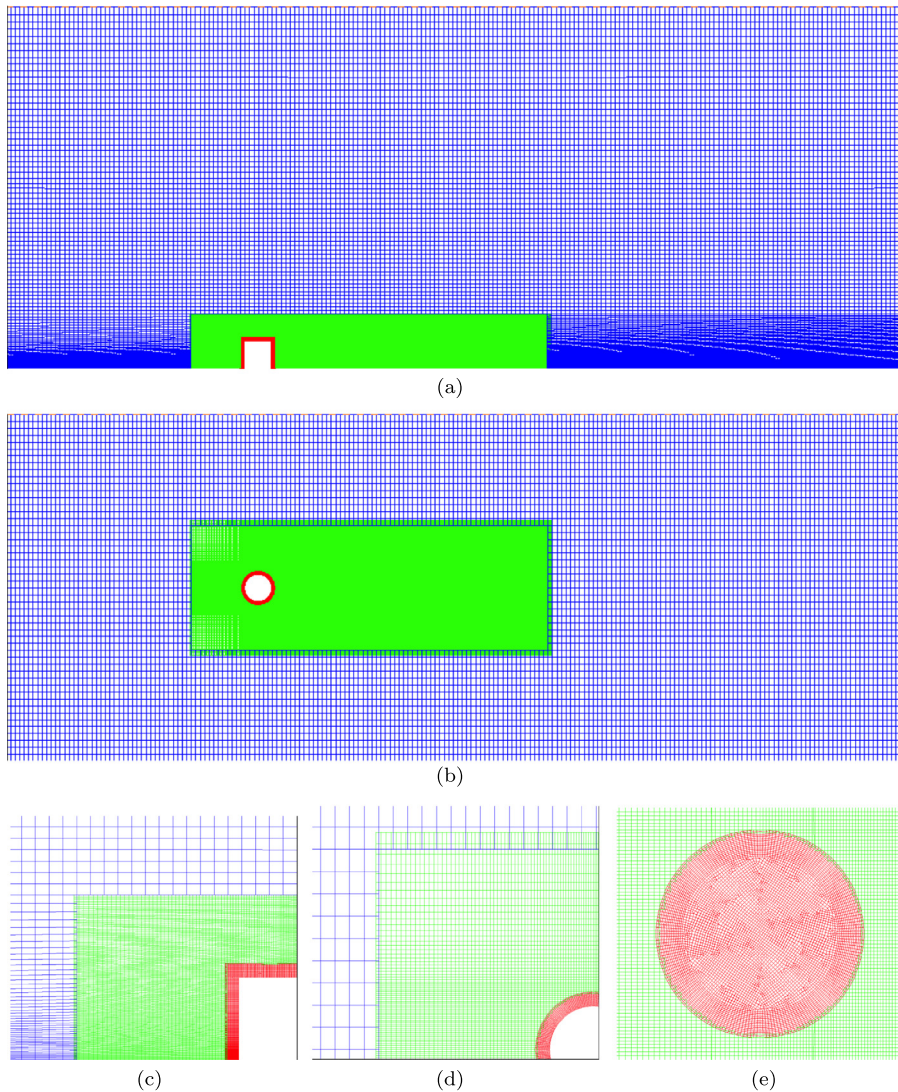


Fig. 23. Views of the computational grids.

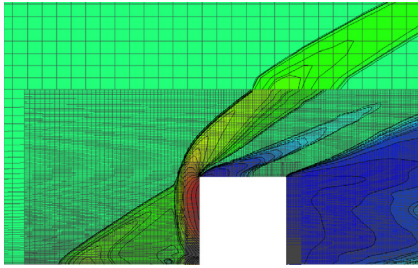
Fig. 24 a shows the isocontours of the time-averaged pressure in the $y = 0$ plane. The shock system is comparable to the one described in [62,63] (sketch provided in Fig. 24 (b)): ahead of the cylinder, the supersonic turbulent boundary layer separates, leading to the formation of a complex three-dimensional lambda-shaped shock wave system. Several phenomena can be observed. First, the strong bow shock is captured in a sharp and non oscillatory way. The weak oblique shock upstream of the cylinder, due to the interaction of the incoming supersonic flow with the turbulent boundary layer, is well resolved. Finally, pressure isolines crossing the boundary between adjacent grid patches, involving drastic grid coarsening and overlapped grids, cross the interface without spurious oscillations or reflections.

Figs. 25 and 26 provide qualitative comparisons of the numerical results with experimental flow visualizations in the xz and xy planes, respectively. The general flow topology and namely the complex shock system and vortical wake are well reproduced.

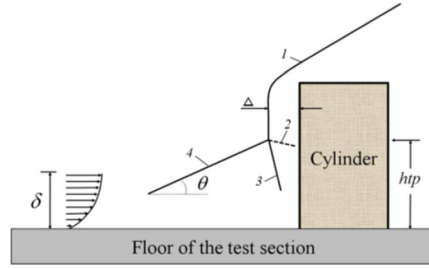
This test case demonstrates the ability of the present methodology in capturing robustly strong shock waves while resolving accurately the most relevant turbulent structures.

6.7. M219 cavity flow

To demonstrate the applicability of the proposed numerical strategy to HRL simulations in complex geometries, we consider the M219 cavity wind-tunnel configuration. This is a three-dimensional cavity mounted into a testing rig. The length of the cavity (L) is 20 inches and the depth (D) and the width (W) are 4 inches. The length-to- depth ratio (L/D) is 5, thus presenting an open cavity flow where the shear layer goes straight across the cavity rather than reattaching on the

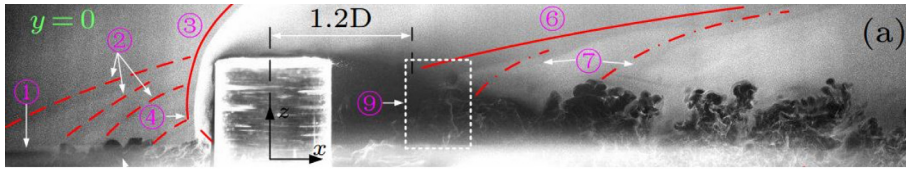


(a) Isocontours of the time averaged pressure next to the stagnation point of the cylinder (in the middle plane $y=0$)



(b) Schematic view of supersonic flow over the cylinder[63]

Fig. 24. Complex shock structures upstream the cylinder.

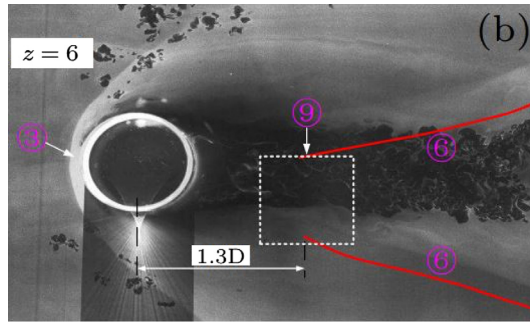


(a) Experiment[62]

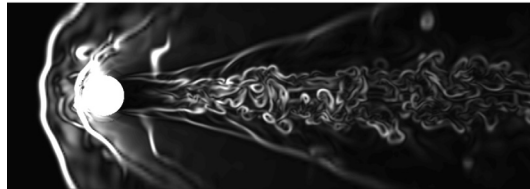


(b) Calculation

Fig. 25. Instantaneous views of the flow in the plane $y = 0$.



(a) Experiment[62]



(b) Calculation

Fig. 26. Instantaneous views of the flow in the plane $z = 6$ mm.

cavity floor. The M219 case has been widely studied since the high Reynolds number subsonic cavity wind-tunnel tests was carried out at QinetiQ [64].

The present computations takes into account the full geometry of the plate and the wind tunnel walls. Fig. 27 provides partial views of the patched grid used to discretize the computational domain.

DDES simulations are carried out on a mesh composed of 6,5 millions cells, concentrated mainly inside the cavity. The y^+ along the wall varies between 20 and 40 (once again, wall functions are applied close to solid walls).

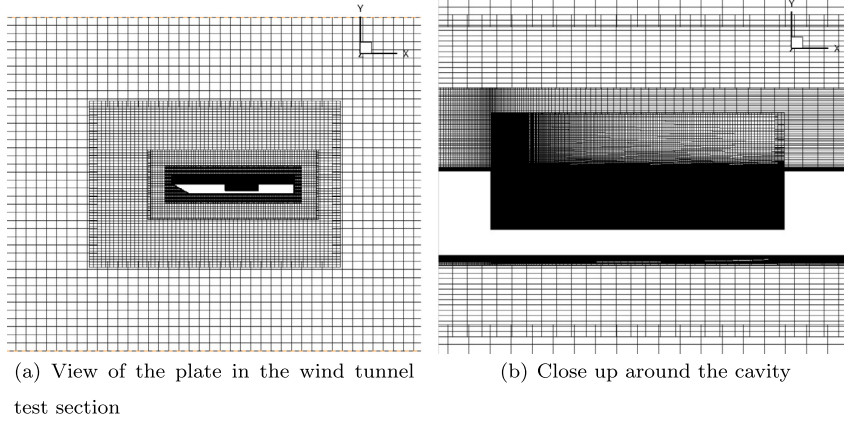


Fig. 27. M219 cavity flow. Partial views of the patched grid used for the computations.

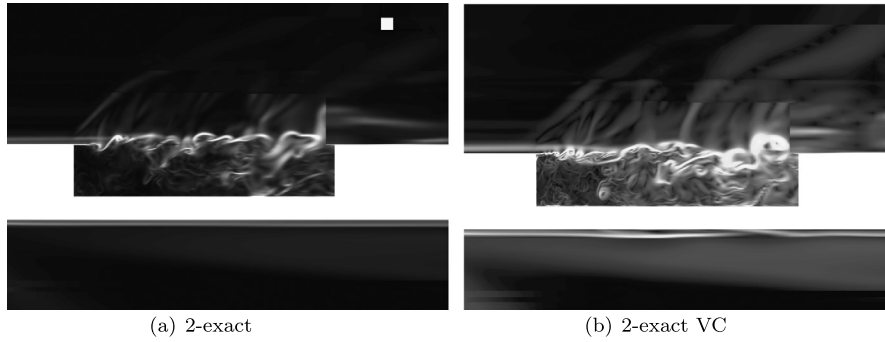


Fig. 28. Supersonic flow past the M219 cavity. Close-up of the numerical Schlieren around the cavity, influence of the Vortex Centering scheme.

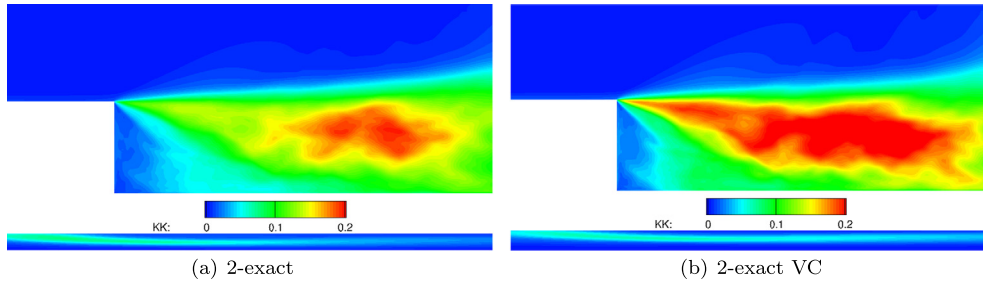


Fig. 29. M219 cavity flow. Normalized resolved kinetic energy K_r .

In the following we present results for a supersonic flow regime, with the Mach number in the test section equal to 1.35.

Fig. 28 shows a close-up of the numerical Schlieren in the cavity region for the 2-exact reconstruction without and with VC. The Kelvin–Helmholtz vortical structures are triggered without delay with the VC scheme at the separation point of the cavity. The flow becomes fully three-dimensional a short distance downstream of the separation point. We can see that the VC scheme solution exhibits more vortical structures, and of smaller size, than the baseline one. **Fig. 29** shows the normalized kinetic energy calculated in the middle plane of the cavity for the baseline and the VC scheme. This is computed as:

$$K_r = \frac{\overline{U_i^2} - \overline{U}^2}{U_{inf}^2} \quad \text{where } \overline{\cdot} \text{ is the time average operator} \quad (98)$$

where U_{inf} is the freestream velocity magnitude and U_i the components of the velocity vector. The VC scheme resolves a larger amount of kinetic energy in the cavity, and especially inside the shear layer, than the baseline. Computations in the subsonic regime (Mach equal to 0.85) have been also carried out, with similar conclusions (not reported for brevity).

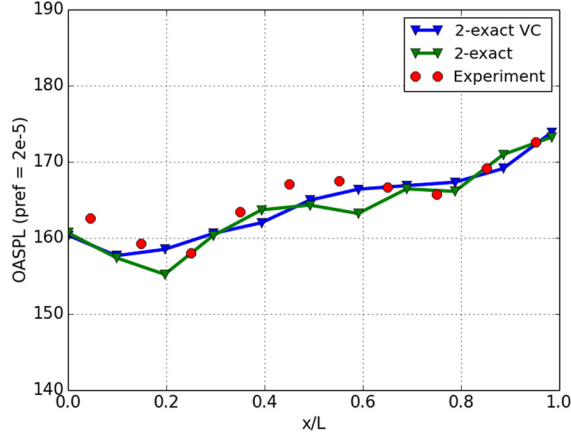


Fig. 30. M219 cavity. Distributions of the Overall Sound Pressure Level (OASPL) for supersonic flows and comparison with the experiments.

Fig. 30 provides a comparison of the computed fluctuating pressure distribution inside the cavity with the experimental data of [64], for the 2-exact reconstruction without and with VC. The results are displayed in terms of the OverAll Sound Pressure Level (OASPL):

$$OASPL = 20 \log \left(\frac{p_{rms}}{p_{ref}} \right), \quad (99)$$

where $p_{ref} = 2 \times 10^{-5}$.

The overall accordance of the present results with the experiments is reasonably good for both the baseline and the VC scheme. The VC scheme provides however a slightly better OASPL.

7. Conclusion

We have presented a general procedure for constructing high-order Godunov-type finite volume methods on general unstructured grids. The procedure efficiently produces a k -exact reconstruction of the primitive variables by recursively correcting the truncation error of a low-order approximation. The proposed successive correction procedure makes use of numerical operators that warrant high accuracy on arbitrary unstructured grids. An analysis of the numerical properties of the proposed family of schemes shows that the number of grid points required to resolve a wavelength with a given error threshold decreases more slowly than the computational cost increases for schemes beyond third-order. In fact, the dispersion error is well controlled over a wide range of wave-numbers when using a scheme of third-order accuracy or higher. Nevertheless, for the proposed robust upwind schemes, the numerical dissipation in the resolved part of the energy spectrum remains excessively high, even when using the highest-order schemes.

In the attempt of ensuring low numerical dissipation in vortex-dominated regions while preserving the shock-capturing capabilities of method and keeping the computational cost as low as possible, a hybrid discretization is developed. The hybrid scheme uses the k -exact Godunov scheme of third-order accuracy, rewritten in such a way that the dissipative part is set to zero according to a local centering criterion. The criterion is based a vortex sensor (Ducros' sensor) to detect vortex-dominated regions, where numerical dissipation needs to be lowered, and satisfies a stability constraint on the grid Reynolds number Re_g . In vortex-dominated regions where the stability constraint is verified, the dissipative part of the scheme is automatically set to zero, and the scheme is locally fourth-order accurate and purely dispersive. In the remaining regions, numerical dissipation is set to the lowest possible value according to stability requirements. Finally, in shock-dominated regions no recentering is applied and the scheme remains fully upwind. The hybrid approach leads to a strong reduction of the numerical dissipation with respect to the baseline scheme without affecting the shock-capturing capabilities of the method and for a negligible increase of the computational cost.

The proposed methodology is applied to a series of tests cases of increasing complexity.

First, numerical results for the well-known Ringleb flow and for the inviscid advection of a vortex prove the capability of the present k -exact schemes of keeping the nominal order of accuracy of the solution on arbitrary grids. The grid topologies used for the numerical tests include smooth and shaken quadrangular grids, stretched quadrangular grids, as well as regular and stretched triangular grids and hybrid grids using overlapping. In all cases, the proposed reconstruction procedure preserves the nominal order of accuracy, even if error levels may be somewhat higher on grids containing triangular elements.

Afterwards, test cases involving the simultaneous computation of shock waves and vortical structures are presented to demonstrate the effectiveness of the hybrid discretization method. Numerical results for a shock/vortex interaction and for a transonic massively separated turbulent flow over a square cylinder prove that the hybrid scheme introduces no dissipation

in well-resolved vortex-dominated regions while capturing the unsteady shock waves sharply and robustly. The scheme is also shown to avoid any unphysical delay in the triggering of Kelvin–Helmoltz instabilities. To demonstrate the capability of the proposed scheme to capture high-Mach number flows with complex shock wave systems, numerical simulations of the supersonic flow at Mach 2.86 past a cylinder mounted on a flat plate were carried out and successfully compared to experimental flow visualizations available from the literature.

As a final application, the hybrid k -exact scheme is used for the RANS/LES simulation of a supersonic flow in a geometrically complex configuration, namely a thick plate with a cavity installed in a wind tunnel. The flow in the cavity is characterized by complex unsteady dynamics. The present calculations succeeded in capturing robustly the shock waves while resolving accurately the most relevant turbulent structures. Results are in good agreement with the available experimental data.

In summary, the proposed hybrid scheme represents a promising candidate for compressible RANS/LES computations in complex geometries of industrial interest. The present methodology has already been applied to detailed analyses of the flow physics about space launchers [65,39,66] by the fluid mechanics department of Airbus Safran Launchers.

Acknowledgements

The authors thank the Association Nationale de la Recherche et de la Technologie (ANRT) and Airbus Safran Launchers for financial support within the framework of the Ph.D number 2011/0934.

Appendix A. Relationship between $\bar{\mathbf{q}}_J$ and $\tilde{\mathbf{q}}_J$

In this appendix we derive expressions relating average values of the primitive variables over one cell, $\bar{\mathbf{q}}_J$, to the average of conservative variables, via vector $\tilde{\mathbf{q}}_J$, i.e. equation (52). These expressions have to be derived independently for each component of $\bar{\mathbf{q}}_J$.

A.1. Relation for the average velocity $\bar{\mathbf{u}}_J$

For the velocity, equation (50) writes:

$$\bar{\mathbf{u}}_J = \mathbf{u}_j + \sum_{m=2}^{n-1} \frac{1}{m!} \mathcal{M}_J^{(m)} : \mathbf{D}^{(m)} \mathbf{u} \Big|_j + \mathcal{O}(h^n) \quad (\text{A.1})$$

On the other hand, average values of the conservative variables can be combined to compute:

$$\tilde{\mathbf{u}}_J = \frac{\bar{\rho} \bar{\mathbf{u}}_J}{\bar{\rho}_J} \quad (\text{A.2})$$

Average values of ρ and $\rho \mathbf{u}$ can be rewritten in terms of the cell-center values using a relation similar to (50). Plugging these developments into (A.2) leads to:

$$\begin{aligned} \tilde{\mathbf{u}}_J &= \mathbf{u}_j \\ &- \frac{\mathbf{u}_j}{\bar{\rho}_J} \sum_{m=2}^{n-1} \frac{1}{m!} \mathcal{M}_J^{(m)} : \mathbf{D}^{(m)} \rho \Big|_j + \frac{1}{\bar{\rho}_J} \sum_{m=2}^{n-1} \frac{1}{m!} \mathcal{M}_J^{(m)} : \mathbf{D}^{(m)} \rho \mathbf{u} \Big|_j + \mathcal{O}(h^n) \end{aligned} \quad (\text{A.3})$$

Subtracting (A.3) from (A.1) and after some algebra, we get:

$$\bar{\mathbf{u}}_J - \tilde{\mathbf{u}}_J = \Delta \mathbf{u}^{m-2} - \frac{1}{\bar{\rho}_J} \mathcal{M}_J^{(m-1)} : \left(\sum_{l=1}^{n-2} \binom{n-1}{l} \mathbf{D}^{(m-1-l)} \rho \Big|_j \otimes \mathbf{D}^{(l)} \mathbf{u} \Big|_j \right) + \mathcal{O}(h^n) \quad (\text{A.4})$$

with $\binom{n-1}{l}$ Newton's binomial and where:

$$\Delta \mathbf{u}^{m-2} = -\frac{1}{\bar{\rho}_J} \left\{ \sum_{m=2}^{n-2} \frac{1}{m!} \mathcal{M}_J^{(m)} : \left(\sum_{l=0}^m \binom{m}{l} \mathbf{D}^{(m-l)} \rho \Big|_j \otimes \mathbf{D}^{(l)} \mathbf{u} \Big|_j - \bar{\mathbf{u}}_J \mathbf{D}^{(m)} \rho \Big|_j - \bar{\rho}_J \mathbf{D}^{(m)} \mathbf{u} \Big|_j \right) \right\}$$

Note that density gradients $\mathbf{D}^{(m)} \rho \Big|_j$ do not need to be reconstructed, since they can be recast in terms of pressure and temperature by using the equation of state:

$$\rho = \frac{P}{RT}$$

and

$$\mathbf{D}^{(m)}\rho|_j = \frac{1}{R} \sum_{l=0}^m \binom{m}{l} \mathbf{D}^{(m-l)}P|_j \otimes \mathbf{D}^{(m)}T^{-1}|_j \quad (\text{A.5})$$

$$= \frac{1}{R} \sum_{l=0}^m \binom{m}{l} \mathbf{D}^{(m-l)}P|_j \otimes \left[(-1)^l l! T_j^{-(l+1)} \mathbf{D}^{(m)}T|_j \right] \quad (\text{A.6})$$

For a 3-exact reconstruction, $\Delta \mathbf{u}^{m-2} = 0$ and (A.4) becomes:

$$\bar{\mathbf{u}}_J - \tilde{\mathbf{u}}_J = -\frac{1}{\bar{\rho}_J} \mathcal{M}_J^{(2)} : \left(\mathbf{D}^{(1)}\rho|_j \otimes \mathbf{D}^{(1)}\mathbf{u}|_j \right) + \mathcal{O}(h^3) \quad (\text{A.7})$$

which requires a 1st-order approximation of the gradients at point \mathbf{x}_j to achieve a third-order approximation.

A.2. Relation for the average pressure \bar{P}_J

We use again a relation of the form (50) to develop \bar{P}_J .
Moreover:

$$\tilde{P}_J = (\gamma - 1) \left(\overline{(\rho E)}_J - \frac{1}{2} \frac{|\overline{(\rho \mathbf{u})}_J|^2}{\bar{\rho}_J} \right) \quad (\text{A.8})$$

Once again, the average conservative variables in equation (A.8) are expanded by means of (50), leading to:

$$\tilde{\mathbf{P}}_J = \mathbf{P}_J \quad (\text{A.9})$$

$$- (\gamma - 1) \frac{\rho_j}{\bar{\rho}_J} \frac{|\mathbf{u}_j|^2}{2} \sum_{m=2}^{n-1} \frac{1}{m!} \mathcal{M}_J^{(m)} : \mathbf{D}^{(m)}\rho|_j \quad (\text{A.10})$$

$$+ (\gamma - 1) \left[\sum_{m=2}^{n-1} \frac{1}{m!} \mathcal{M}_J^{(m)} : \left(\mathbf{D}^{(m)}\rho E|_j - \frac{1}{2\bar{\rho}_J} \mathbf{D}^{(m)}|\rho \mathbf{u}|^2|_j \right) \right] + \mathcal{O}(h^n) \quad (\text{A.11})$$

With a procedure similar to that used for \mathbf{u} , we finally get:

$$\bar{\mathbf{P}}_J - \tilde{\mathbf{P}}_J = \Delta \mathbf{P}^{n-2} - \frac{(\gamma - 1) \mathcal{M}_J^{(m-1)}}{\bar{\rho}_J (n-1)!} : \left[\sum_{l=1}^{n-2} \binom{n-1}{l} \mathbf{D}^{(m-1-l)}\rho|_j \otimes \mathbf{D}^{(m)}|\rho \mathbf{u}|^2|_j \right] + \mathcal{O}(h^n) \quad (\text{A.12})$$

with:

$$\Delta \mathbf{P}^{n-2} = - \sum_{m=2}^{n-2} \frac{1}{m!} \mathcal{M}_J^{(m)} : \left\{ \frac{(\gamma - 1)}{\bar{\rho}_J} \left[\bar{\rho}_J \mathbf{D}^{(m)}(\rho E)|_j \right. \right. \quad (\text{A.13})$$

$$\left. \left. + \rho_j \frac{|\mathbf{u}_j|^2}{2} \mathbf{D}^{(m)}\rho|_j - \frac{1}{2} \mathbf{D}^{(m)}|\rho \mathbf{u}|^2|_j \right] - \mathbf{D}^{(m)}P|_j \right\} \quad (\text{A.14})$$

Product derivatives like $\mathbf{D}^{(m)}(\rho E)|_j$ or $\mathbf{D}^{(m)}|\rho \mathbf{u}|^2|_j$ can be developed in terms of the primitive variables and of the density ρ . Precisely:

$$\mathbf{D}^{(m)}|\rho \mathbf{u}_j|^2|_j = \sum_{l=0}^m \binom{m}{l} \mathbf{D}^{(l)}\rho|_j \otimes \mathbf{D}^{(l)}|\mathbf{u}_j|^2|_j \quad (\text{A.15})$$

and

$$\mathbf{D}^{(m)}(\rho E)|_j = \sum_{l=0}^m \binom{m}{l} \mathbf{D}^{(m-l)}\rho|_j \otimes \mathbf{D}^{(l)}E|_j \quad (\text{A.16})$$

$$= \sum_{l=0}^m \binom{m}{l} \mathbf{D}^{(m-l)}\rho|_j \otimes \left[C_v \mathbf{D}^{(l)}T|_j + \frac{1}{2} \mathbf{D}^{(m)}|\mathbf{u}|^2|_j \right] \quad (\text{A.17})$$

Furthermore,

$$\mathbf{D}^{(m)} | \mathbf{u}_j |^2 \Big|_j \sum_{l=0}^m \binom{m}{l} \mathbf{D}^{(m-l)} \mathbf{u} \Big|_j \otimes \mathbf{D}^{(l)} \mathbf{u}_j \Big|_j \quad (\text{A.18})$$

For a 3-exact reconstruction:

$$\bar{\mathbf{P}}_J - \tilde{\mathbf{P}}_J = -\frac{(\gamma-1)}{\bar{\rho}_J} \mathcal{M}_J^{(2)} : \left\{ \mathbf{D}^{(1)} \rho \Big|_j \otimes \left[\mathbf{D}^{(1)} \rho \Big|_j \otimes | \mathbf{u}_j |^2 + 2\rho_j \mathbf{D}^{(1)} | \mathbf{u} \Big|_j \right] \right\} + \mathcal{O}(h^3) \quad (\text{A.19})$$

A.3. Relation for the average pressure \bar{T}_J

We proceed again as in the two preceding cases. This time, we develop the average conservative variables in:

$$\tilde{T} = \frac{\tilde{P}}{R\bar{\rho}} \quad (\text{A.20})$$

After some developments, is it possible to prove that:

$$\begin{aligned} \bar{T}_J - \tilde{T}_J &= \Delta \mathbf{T}^{n-2} \\ &\quad - \frac{1}{\bar{\rho}_J} \frac{1}{(n-1)!} \mathcal{M}_J^{(n-1)} : \left\{ \sum_{l=1}^{n-2} \binom{n-1}{l} \mathbf{D}^{(m-1-l)} T \Big|_j \otimes \left[\mathbf{D}^{(l)} \rho \Big|_j - \frac{1}{2c_v \bar{\rho}_J} \mathbf{D}^{(l)} \rho | \mathbf{u} |^2 \Big|_j \right] \right\} + \mathcal{O}(h^n) \end{aligned} \quad (\text{A.21})$$

$$\Delta \mathbf{T}^{n-2} = - \sum_{m=2}^{n-2} \frac{1}{m!} \mathcal{M}_J^{(m)} : \left\{ \frac{-T_j}{\bar{\rho}_J} \mathbf{D}^{(m)} \rho \Big|_j + (\gamma-1) \frac{\rho_j}{\bar{\rho}_J} \frac{| \mathbf{u} |^2_j}{\bar{\rho}_J} \mathbf{D}^{(m)} \rho \Big|_j + \frac{(\gamma-1)}{\bar{\rho}_J R} \mathbf{D}^{(m)} P \Big|_j - \mathbf{D}^{(m)} T \Big|_j \right\} \quad (\text{A.22})$$

Again, $\Delta \mathbf{T}^{n-2}$ is null for $n=3$. Thus, for the 3rd-order reconstruction:

$$\bar{T}_J - \tilde{T}_J = -\frac{1}{\bar{\rho}_J} \mathcal{M}_J^{(2)} : \left\{ \mathbf{D}^{(1)} \rho \Big|_j \otimes \left[\mathbf{D}^{(1)} T \Big|_j - \frac{1}{2c_v \bar{\rho}_J} \mathbf{D}^{(1)} \rho | \mathbf{u} |^2 \Big|_j \right] \right\} + \mathcal{O}(h^n) \quad (\text{A.23})$$

A.4. Numerical approximation of $\bar{\mathbf{q}}_J - \tilde{\mathbf{q}}_J$

After development of all product derivatives, expressions in equation (A.4), (A.12) and (A.21), only on the volume moments of order $n-1$ and derivatives of the primitive variables of order $n-2$. Thus, n th-order accuracy is achieved by replacing exact derivatives of order $m \leq n-2$ with a numerical approximation of order $n-m-1$.

Appendix B. Choice of the integration point Γ for surface integrals

In Section 3.3.1 we introduced a one-point integration formula for the surface integrals on cell faces, based on the reconstruction of the flux \mathbf{f} and of its derivatives at a suitable point Γ . When the surfaces are flat, the more natural choice for Γ is the center of figure of the face A_{JK} , since it sets to zero the first-order error terms in the integration formula (48). For general skew interfaces, it is not the case. In this case. Then integration point Γ is chosen in such a way as to minimize the first order truncation error term of (48). For this purpose, consider the formula written up to the second order terms:

$$\iint_{A_{JK}} \mathbf{F} \cdot \mathbf{n} dS = \mathbf{F}|_{\Gamma} \cdot \mathcal{S}_{A_{JK}}^{(0)} + \underbrace{\mathbf{D}^{(1)} \mathbf{F} \Big|_{\Gamma}}_{\mathcal{E}_{\Gamma}} \cdot \mathcal{S}_{A_{JK}}^{(1)} + |A_{JK}| \mathcal{O}(h^2). \quad (\text{B.1})$$

Similar expressions can be written for any other integration point b in A_{JK} , just by replacing the index Γ in the equation above with b . The flux at point Γ can be Taylor-series expanded around the new point b :

$$\mathbf{F}|_{\Gamma} = \mathbf{F}|_b + \mathbf{D}^{(1)} \mathbf{F} \Big|_b \cdot (\mathbf{x}_{\Gamma} - \mathbf{x}_b) + \mathcal{O}(h^2). \quad (\text{B.2})$$

Plugging the development (B.2) in equation (B.1):

$$\iint_{A_{JK}} \mathbf{F} \cdot \mathbf{n} dS = \mathbf{F}|_b \cdot \mathcal{S}_{A_{JK}}^{(0)} + \underbrace{\mathbf{D}^{(1)} \mathbf{F} \Big|_{\Gamma} \cdot \mathcal{S}_{A_{JK}}^{(1)} + \left(\mathbf{D}^{(1)} (\mathbf{F}) \Big|_b \cdot (\mathbf{x}_{\Gamma} - \mathbf{x}_b) \right) \cdot \mathcal{S}_{A_{JK}}^{(0)}}_{\mathcal{E}_b} \quad (\text{B.3})$$

$$+ |A_{JK}| \mathcal{O}(h^2) \quad (\text{B.4})$$

The square norm of the first-order error \mathcal{E}_b can be expressed as a function of the integration error \mathcal{E}_{Γ} :

$$\begin{aligned}
|\mathcal{E}_{\mathbf{b}}|^2 &= |\mathcal{E}_{\Gamma}|^2 \\
&+ \left(\left(\mathbf{D}^{(1)}(\mathbf{F}) \Big|_b \cdot (\mathbf{x}_{\Gamma} - \mathbf{x}_b) \right) \cdot \mathcal{S}_{AJK}^{(0)} \right)^2 \\
&+ 2 \left(\mathbf{D}^{(1)}\mathbf{F} \Big|_{\Gamma} \cdot \mathcal{S}_{AJK}^{(1)} \right) \cdot \left(\left(\mathbf{D}^{(1)}\mathbf{F} \Big|_b \cdot (\mathbf{x}_{\Gamma} - \mathbf{x}_b) \right) \cdot \mathcal{S}_{AJK}^{(0)} \right) \\
&+ |A_{JK}| O(h^2)
\end{aligned} \tag{B.5}$$

The third term on the right-hand side is a null vector if the coordinates of the integration point Γ are chosen in such a way that the second-order tensor/vector product $\mathcal{S}_{AJK}^{(1)} \cdot \mathcal{S}_{AJK}^{(0)} = \mathbf{0}$ since:

$$|\mathcal{E}_{\mathbf{b}}|^2 = |\mathcal{E}_{\Gamma}|^2 + \left(\left(\mathbf{D}^{(1)}(\mathbf{F}) \Big|_b \cdot (\mathbf{x}_{\Gamma} - \mathbf{x}_b) \right) \cdot \mathcal{S}_{AJK}^{(0)} \right)^2 + |A_{JK}| O(h^2) \tag{B.6}$$

so that

$$|\mathcal{E}_{\mathbf{b}}|^2 \geq |\mathcal{E}_{\Gamma}|^2 \quad \forall \mathbf{x}_b \in A_{JK} \tag{B.7}$$

In practice, the coordinates of point Γ are found numerically by solving the linear system:

$$\mathcal{S}_{AJK}^{(1)} \cdot \mathcal{S}_{AJK}^{(0)} = \mathbf{0} \quad \text{i.e.} \quad \left(\iint_{A_{JK}} (\mathbf{x} - \mathbf{x}_{\Gamma}) \otimes \mathbf{n} dS \right) \cdot \mathcal{S}_{AJK}^{(0)} = \mathbf{0} \tag{B.8}$$

References

- [1] C.K. Tam, J.C. Webb, Dispersion-relation-preserving finite difference schemes for computational acoustics, *J. Comput. Phys.* 107 (1993) 262–281.
- [2] S. Pirozzoli, On the spectral properties of shock-capturing schemes, *J. Comput. Phys.* 219 (2006) 489–497.
- [3] D. Fauconnier, C.D. Langhe, E. Dick, Construction of explicit and implicit dynamic finite difference schemes and application to the large-eddy simulation of the Taylor–Green vortex, *J. Comput. Phys.* 228 (2009) 8053–8084.
- [4] K. Grimich, P. Cinnella, A. Lerat, Spectral properties of high-order residual-based compact schemes for unsteady compressible flows, *J. Comput. Phys.* 252 (2013) 242–262.
- [5] M.R. Visbal, D.V. Gaitonde, High-order accurate methods for complex unsteady subsonic flows, *AIAA J.* 37 (1999) 1231–1239.
- [6] P. Castonguay, D. Williams, P. Vincent, M. Lopez, A. Jameson, On the Development of a High-Order, Multi-GPU Enabled, Compressible Viscous Flow Solver for Mixed Unstructured Grids, *Tech. rep., AIAA Paper 2011-3229*, 2011.
- [7] A. Mahammad, Z. Wang, C. Lian, Large eddy simulation of flow over a cylinder using high-order spectral difference method, *Adv. Appl. Math. Mech.* 2 (2011) 451–466.
- [8] P.R. Spalart, Strategies for turbulence modelling and simulations, *Int. J. Heat Fluid Flow* 21 (2000) 252–263.
- [9] P.R. Spalart, W.-H. Jou, M. Strelets, S. Allmaras, Comments on the feasibility of LES for wings and on the hybrid RANS/LES approach, in: *Advances in DNS/LES, Proceedings of the First AFOSR International Conference on DNS/LES*, Ruston, Louisiana, 1997.
- [10] F. Menter, Y. Ergov, The scale-adaptive simulation method for unsteady turbulent flow predictions. Part 1: theory and model description, *Flow Turbul. Combust.* 25 (85) (2010) 113–138.
- [11] S.S. Girimaji, Partially-averaged Navier–Stokes model for turbulence: a Reynolds-averaged Navier–Stokes to direct numerical simulation bridging method, *J. Appl. Mech.* 73 (2006) 413–421.
- [12] J. Perot, J. Gadebush, A self-adapting turbulence model for flow simulation at any mesh resolution, *Phys. Fluids* 19 (2007) 1–11.
- [13] P. Sagaut, S. Deck, M. Terracol, *Multiscale and Multiresolution Approches in Turbulence*, Imperial College Press, 2006.
- [14] P.R. Spalart, Detached-eddy simulation, *Annu. Rev. Fluid Mech.* 41 (2009) 181–202.
- [15] Z. Wang, High-order methods for the Euler and Navier–Stokes equations on unstructured grids, *Prog. Aerosp. Sci.* 43 (2007) 1–41.
- [16] Z.J. Wang, K. Fidkowski, R. Abgrall, F. Bassi, D. Caraeni, A.C.N. Kroll, High-order CFD method: current status and perspective, *Int. J. Numer. Methods Fluids* 72 (2013) 811–845.
- [17] N. Kroll, C. Hirsch, F. Bassi, C. Johnston, K. Hillewaert (Eds.), *IDIHOM: Industrialization of High-Order Methods – a Top-Down Approach*, Springer, 2014.
- [18] S. Godunov, A difference scheme for numerical computation of discontinuous solutions of equations of fluid dynamics, *Sb. Math.* 47 (1959) 271–306.
- [19] B. van Leer, Towards the ultimate conservative difference scheme. V. A second-order sequel to Godunov’s method, *J. Comput. Phys.* 32 (1979) 101–136.
- [20] T.J. Barth, P. Frederickson, Higher Order Solution of the Euler Equations on Unstructured Grids Using Quadratic Reconstruction, *Tech. rep., AIAA Paper 90-0013*, 1990.
- [21] T.J. Barth, Recent Developments in High Order k-Exact Reconstruction on Unstructured Meshes, *Tech. rep., AIAA Paper 93-0668*, 1993.
- [22] P. Vankeirsbilck, H. Deconinck, Higher order upwind finite volume schemes with ENO-properties for general unstructured meshes, *AGARD Report R-787*, ISBN 92-835-0671-5.
- [23] C. Ollivier-Gooch, C. Van Altena, A high-order-accurate unstructured mesh finite-volume scheme for the advection diffusion equation, *J. Comput. Phys.* 181 (2002) 729–752.
- [24] D. Caraeni, D. Hill, Efficient 3rd-Order Finite-Volume Discretization Using Iterative Quadratic Data Reconstruction on Unstructured Grids, *Tech. rep., AIAA Paper 2009-1332*, 2009.
- [25] F. Haider, P. Brenner, B. Courbet, J.P. Croisille, Efficient implementation of high order reconstruction in finite volume methods, in: *Finite Volumes for Complex Application VI—Problem & Perspectives*, in: Springer Proceedings in Mathematics, vol. 4, 2011, pp. 553–560.
- [26] L. Ivan, C.P. Groth, High-order solution-adaptive Central Essentially Non-Oscillatory (CENO) method for viscous flows, *J. Comput. Phys.* 257 (Part A (0)) (2014) 830–862.
- [27] H. Yang, Z. Chen, A. Przekwas, J. Dudley, A high-order CFD method using successive differentiation, *J. Comput. Phys.* 281 (0) (2015) 690–707.
- [28] F. Haider, *Discrétisation en maillage non structuré général et application LES*, Ph.D. thesis, UPMC, 2009.
- [29] M. Dumbser, M. Käser, V.A. Titarev, E.F. Toro, Quadrature-free non-oscillatory finite volume schemes on unstructured meshes for nonlinear hyperbolic systems, *J. Comput. Phys.* 226 (1) (2007) 204–243.
- [30] D. Mavriplis, Revisiting the Least-Squares Procedure for Gradient Reconstruction on Unstructured Meshes, *Tech. rep., AIAA Paper 2003-3986*, 2003.

- [31] E. Shima, K. Kitamura, H. Takanori, Green-Gauss/weighted-least-squares hybrid gradient reconstruction for arbitrary polyhedra unstructured grids, *AIAA J.* 51 (2013) 2740–2747.
- [32] E. Garnier, M. Mossi, P. Sagaut, P. Comte, M. Deville, On the use of shock-capturing schemes for large-eddy-simulation, *J. Comput. Phys.* 153 (1999) 273–311.
- [33] D. Drikakis, Advances in turbulent flow computations using high-resolution methods, *Prog. Aerosp. Sci.* 39 (2003) 412–424.
- [34] A. Travin, M. Shur, M. Strelets, P.R. Spalart, Physical and numerical upgrades in the detached-eddy simulation of complex turbulent flows, in: R. Friedrich, W. Rodi (Eds.), *Fluid Mechanics and Its Applications*, in: *Advances in LES of Complex Flows*, vol. 65, Springer, Netherlands, 2002, pp. 239–254.
- [35] I. Mary, P. Sagaut, Large eddy simulation of flow around an airfoil near stall, *AIAA J.* 40 (6) (2002) 1139–1145.
- [36] P.R. Spalart, S. Deck, S. Shur, M. Squires, M. Strelets, A. Travin, A new version of detached-eddy simulation, resistant to ambiguous grid densities, *Theor. Comput. Fluid Dyn.* 20 (2006) 181–195.
- [37] N. Chauvet, S. Deck, L. Jacquin, Zonal-detached-eddy simulation of a controlled propulsive jet, *AIAA J.* 45 (10) (2007) 2458–2473.
- [38] S. Catris, B. Aupoix, Density corrections for turbulence models, *Aerosp. Sci. Technol.* 4 (2000) 1–11.
- [39] G. Pont, *Self Adaptive Turbulence Models for Unsteady Compressible Flows*, Ph.D. thesis, Arts et Métiers ParisTech, 2015.
- [40] P. Brenner, Unsteady flows about bodies in relative motion, in: *1st AFOSR Conference on Dynamic Motion CFD Proceedings*, Rutgers University, New Jersey, USA, 1996.
- [41] R.A. Gentry, R.E. Martin, B.J. Daly, An Eulerian differencing method for unsteady compressible flow problems, *J. Comput. Phys.* 1 (1966) 87–118.
- [42] G. Aubard, P. Stefanin Volpiani, X. Gloerfelt, J.-C. Robinet, Comparison of subgrid-scale viscosity models and selective filtering strategy for large-eddy simulations, *Flow Turbul. Combust.* 91 (3) (2013) 497–518.
- [43] X. Gloerfelt, P. Cinnella, Investigation of high-order methods in large-eddy simulation of separated flow in a channel with periodic constrictions, *Direct and Large Eddy Simulation X, ERCOFTAC Series*.
- [44] B. Manguerra, *Méthodes de volumes finis d'ordre élevé en maillage non coincident pour les écoulements dans les turbomachines*, Ph.D. thesis, Arts et Métiers ParisTech, 2016.
- [45] L. Takacs, A two-step scheme for the advection equation with minimized dissipation and dispersion error, *Mon. Weather Rev.* 113 (1985) 1050–1065.
- [46] G. Aubard, P. Stefanin Volpiani, X. Gloerfelt, J.-C. Robinet, Comparison of subgrid-scale viscosity models and selective filtering strategy for large eddy simulations, *Flow Turbul. Combust.* 91 (3) (2013) 497–518.
- [47] F. Ducros, V. Ferrand, F. Nicoud, C. Weber, D. Darracq, Large eddy simulation of the shock/turbulence interaction, *J. Comput. Phys.* 152 (1999) 517–549.
- [48] F. Ringleb, *Exakte Lösungen der Differentialgleichungen einer adiabatischen Gasströmung*, *J. Appl. Math. Mech.* 20 (1940) 185–198.
- [49] H. Lamb, *Hydrodynamics*, University Press, Cambridge [Eng.], 1932.
- [50] C. Shu, *Essentially Non-Oscillatory and Weighted Essentially Non-Oscillatory Schemes for Hyperbolic Conservation Laws*, ICASE Report 97-65 1997.
- [51] M.E. Brachet, D. Meiron, S. Orszag, B. Nickel, R. Morf, U. Frisch, The Taylor-Green vortex and fully developed turbulence, *J. Stat. Phys.* 34 (5–6) (1984) 1049–1063.
- [52] S. Hickel, N.A. Adams, J.A. Domaradzki, An adaptive local deconvolution method for implicit LES, *J. Comput. Phys.* 213 (2006) 413–436.
- [53] D. Fauconnier, C. Bogey, E. Dick, On the performance of relaxation filtering for large-eddy simulation, *J. Turbul.* 14 (1) (2013) 22–49.
- [54] L. Diosady, S. Murman, Case 3.3: Taylor Green Vortex Evolution, Tech. rep., 3rd International Workshop on High-Order CFD Methods, January 3–4, 2015, Kissimmee, FL, 2015.
- [55] T. Nakagawa, Vortex shedding behind a square cylinder in transonic flows, *J. Fluid Mech.* 178 (1987) 303–323.
- [56] G. Pont, P. Cinnella, J.-C. Robinet, P. Brenner, Assessment of automatic hybrid RANS/LES models for industrial CFD, in: *AIAA Paper 2014-2691*, 2014.
- [57] G. Pont, P. Cinnella, J.-C. Robinet, P. Brenner, Automatic hybrid RANS/LES strategy for industrial CFD, in: *5th Symposium on Hybrid RANS-LES Methods*, 2014.
- [58] P. Brenner, Three-dimensional aerodynamics with moving bodies applied to solid propellant, in: *27th Joint Propulsion Conference*, Sacramento, USA, AIAA Paper 1991-2304.
- [59] P. Brenner, Simulation du mouvement relatif de corps soumis à un écoulement instationnaire par une méthode de chevauchement de maillage, in: *AGARD Conference Proceedings 578: Progress and Challenges in CFD Methods and Algorithms*, Seville, Spain, 1995.
- [60] P. Brenner, A conservative overlapping grid method to simulate rocket stage separation, in: *3rd Symposium on Overset Composite Grid and Solution Technology*, Los Alamos, 1996.
- [61] A. Prasard, C.H.K. Williamson, The instability of the shear layer separating from a bluff body, *J. Fluid Mech.* 333 (1996) 375–402.
- [62] D. Wang, Y. Zhao, Z. Xia, Q. Wang, Z. Luo, Flow visualization of supersonic flow over a finite cylinder, *Chin. Phys. Lett.* 29 (8) (2012) 084702.
- [63] D.-D. Gang, S.-he Yi, L. He, Characteristics of the cylinder-induced shock wave and turbulent boundary layer interactions, *J. Vis.* 19 (4) (2016) 581–585.
- [64] M. Henshaw, M219 cavity case. Verification and validation data for computational unsteady aerodynamics, Defense Technical Information Center, ADP010729.
- [65] G. Pont, D. Puech, P. Brenner, Hybrid RANS/LES simulation of a space launcher using a high order finite volume scheme and grid intersections technique, in: *6th Symposium on Hybrid RANS-LES Methods*, 2016.
- [66] G. Pont, P. Brenner, P. Cinnella, J. Robinet, High-order hybrid RANS/LES strategy for industrial applications, *Direct and Large Eddy Simulation X, ERCOFTAC Series*.