Journal of Computational Physics 457 (2022) 111043

Contents lists available at ScienceDirect

Journal of Computational Physics

www.elsevier.com/locate/jcp

On the properties of high-order least-squares finite-volume schemes

G. Saez-Mischlich [a,*], J. Sierra-Ausin [b], G. Grondin [a], J. Gressier [a]

[a] Institut Supérieur de l'Aéronautique et de l'Espace (ISAE-SUPAERO), Toulouse 31400, France
[b] Institut de Mécanique des Fluides de Toulouse (IMFT), Toulouse 31400, France

ARTICLE INFO

Article history:
Received 2 March 2021
Received in revised form 28 December 2021
Accepted 2 February 2022
Available online 8 February 2022

Keywords:
Finite-volume
High-order
Least-squares
k-exact

ABSTRACT

High-order finite-volume schemes based on polynomial least-squares methods are an active research topic for the discretization of hyperbolic equations as they allow to obtain high-order spatial discretization schemes in arbitrary meshes. However, few studies have analyzed their performance in good-quality/near-to-uniform meshes, which are commonly used as a meshing strategy in zones where turbulent effects are important. In this paper, the theoretical numerical properties of commonly used least-squares (LSQ) k-exact high-order finite volume schemes are studied in one-dimensional and in several two-dimensional meshes (with some remarks regarding their properties in three-dimensional meshes). These results are compared to those obtained using fully-constrained polynomial reconstructions only compatible with structured meshes. The numerical properties of the schemes are investigated through the von Neumann analysis methodology applied to the one-dimensional and two-dimensional finite volume formulation, including temporal discretization errors. This analysis is also extended to non-uniform and unstructured two-dimensional meshes. At last, the schemes are tested with several numerical experiments using the linear advection, the Euler equations and the Navier-Stokes equations. The analysis of both studies yields similar conclusions regarding the numerical errors and stability of the different studied schemes showing that the high-order least-squares finite volume schemes yield stable and robust results across different uniform and non-uniform unstructured meshes. However, their performance is heavily degraded in simulations with low mesh resolution compared to schemes specially catered to structured meshes. On the other hand, the latter schemes lack stability and robustness in general structured meshes and its formulation cannot be straightforwardly extended to unstructured meshes. Moreover, this work shows that the use of weighted-LSQ can drastically improve the results of LSQ schemes in under-resolved simulations.

© 2022 Elsevier Inc. All rights reserved.

1. Introduction

Finite volume (FV) is a well-established numerical technique for the reconstruction of solutions of hyperbolic partial differential equations (PDEs), e.g. Euler's and Maxwell's equations. Its success relies on two key properties: these schemes fulfill the discrete global conservation property and L_∞–stable schemes are relatively easy to design, e.g. non-oscillatory

[*] Corresponding author.
E-mail address: g.saez-mischlich@isae-supaero.fr (G. Saez-Mischlich).

https://doi.org/10.1016/j.jcp.2022.111043
0021-9991/© 2022 Elsevier Inc. All rights reserved.

methods (ENO [1], WENO [2,3], MOOD [4]). Finite volume is a particular case of a Bubnov-Galerkin method, i.e. test and solution functions do not belong to the same functional space, where test functions are constant along each element and solution functions are polynomial in each element. Therefore, higher-order reconstructions are possible if one considers polynomial reconstructions of higher degree. In particular, the quality of a finite volume reconstruction depends on two operations: variable reconstruction from neighboring element-averaged and face average normal flux reconstruction. The former is usually realized by interpolation and it can lead to the well-known k-exact schemes [2,5–7], whereas the latter is commonly evaluated in high-order formulations via quadrature integration [8–11]. There exist other face average normal flux reconstruction methods based on successive error corrections via Taylor expansion series. Interested reader is referred to the studies of [12–14] for more information on these methods.

Connectivity characteristics of the discretization of the physical domain play an important role in the numerical properties of finite volume schemes. Structured numerical grids possess a constant number of face and vertex neighbors to each element and there exists a labeling function relating N-tuples of indexes to element centroids, e.g. in two dimensional domains, there is a labeling $(i, j) \in \mathbb{Z}^2$ for each element centroid. Under certain conditions, these properties allow to determine coefficients of the reconstruction scheme by solving a non-singular linear system of equations [15,16]. Nonetheless, unstructured grids do not have, in general, a constant number of neighbors for each element and the labeling of each element centroid cannot be decomposed onto a Cartesian product of N-sets of indexes for N-dimensional domains. For such reason, the stencils employed for the reconstruction of the solution are not unique and its selection is crucial in the numerical properties of the scheme [6,17,18]. The determination of the variable reconstruction at the faces may be determined by an iterative procedure of successive Green–Gauss [12,14] approximations or by the selection of a stencil and then solving a L^2-minimization problem using a least-squares procedure [6,9,2].

The present study aims to analyze numerical properties, e.g. order of accuracy, stability, dispersion and diffusion errors, of finite volume schemes on structured and unstructured grids with a quadrilateral discretization applied to hyperbolic problems with smooth solutions. In particular the emphasis is on properties of least-squares schemes with respect to their structured counter-part. Firstly, the general framework of finite volume method is synthesized in Section 2. Then in Section 3 the attention is paid on a non-compact reconstruction method on unstructured grids by a least-squares procedure with a given stencil based on the superposition of topological layers of neighbors. Such a methodology is particularized in Section 3.2 for structured grids. An analytical study of the truncation error, i.e. order of accuracy, is carried out in Section 4.1 for one and two-dimensional structured grids. Additionally, a numerical test based on the k-polynomial reconstruction property is employed to determine the order of accuracy of least-squares schemes on unstructured grids. Spectral properties of schemes such as – L^2-stability, dispersive and diffusion errors – are studied in Section 5. Finally, a set of numerical examples is reported in Section 6 to validate the analytical findings.

2. Finite volume applied to conservation laws

Finite volume methods naturally apply to conservation laws expressed in divergence form:

$$\frac{\partial u(\boldsymbol{x}, t)}{\partial t} + \nabla \cdot \mathcal{F}(u, \nabla u) = 0 \quad \forall \boldsymbol{x} \text{ in } \Omega \subset \mathbb{R}^{N_d}, \quad N_d = 1, 2, 3 \tag{1}$$

where $u(\boldsymbol{x}, t)$ is a conserved scalar or vector quantity, $\boldsymbol{x}$ is the position vector, $\mathcal{F}$ is the flux operator and N_d is the number of spatial dimensions of the problem. The flux operator is usually decomposed in two components: the convective flux $\mathcal{F}_c(u)$ which depends only on the conserved variable and not its gradient, and the viscous flux $\mathcal{F}_v(u, \nabla u)$ which depends both on the conserved variable and its gradient. The numerical resolution of Eq. (1) is carried out via the finite volume method.

2.1. Partitioning of the physical domain

Several types of unstructured meshes are common in finite volume methods. Most polygonal meshes are composed of N_d-simplexes (triangles or tetrahedrons) and tensor-product elements (quadrilaterals or hexahedrons). In this work the meshes will be decomposed in $N_{\mathcal{V}}$ linear tensor-product elements or cells. As it is common in the literature, independently of the shape of the chosen polygon covering the physical domain Ω, the partitioning of the numerical domain Ω_h produces the partition $\boldsymbol{V}_h$, which possesses the usual required properties for finite element method, see [19]. Cells $\mathcal{V}_i \in \boldsymbol{V}_h$ are closed convex volume elements, satisfying the following conformal property:

$$\begin{aligned} \text{if } \mathcal{V}_i, \mathcal{V}_j \in \boldsymbol{V}_h, \quad & \text{then either } \mathcal{V}_i \cap \mathcal{V}_j = \varnothing, \\ & \text{or } \mathcal{V}_i \cap \mathcal{V}_j \text{ is a common} \begin{cases} \text{vertex,} \\ \text{ edge} \\ \text{or face} \end{cases} \text{of } \mathcal{V}_i \text{ and } \mathcal{V}_j, \\ & \text{or } \mathcal{V}_i = \mathcal{V}_j. \end{aligned} \tag{2}$$

In addition, let us consider the characteristic length h_i of each volume element, which proves to be useful in the analysis of numerical properties of the space discretization. This parameter is defined [20] as

$$h_i = \frac{|\mathcal{L}_i|}{\sum_{\ell \in \mathcal{L}_i} |\mathcal{S}_\ell|} |\mathcal{V}_i| \quad , \tag{3}$$

where $\mathcal{L}_i$ is the set of indices of the faces $\mathcal{S}_\ell$ of element $\mathcal{V}_i$, with respective cardinality $|\mathcal{L}_i|$ and measures $|\mathcal{S}_\ell|$ and $|\mathcal{V}_i|$. The global edge size h is defined as the minimum edge size

$$h = \min_i h_i \quad . \tag{4}$$

Finally, the centroid $\boldsymbol{x}_i$ of each element is defined as

$$\boldsymbol{x}_i = \frac{1}{|\mathcal{V}_i|} \int_{\mathcal{V}_i} \boldsymbol{x} \, \mathrm{d}\mathcal{V} \quad . \tag{5}$$

2.2. Finite volume semi-discretization

The finite volume formulation of Eq. (1) describes the evolution of the cell average $\overline{u}_i$ of the conserved quantity on cell $\mathcal{V}_i$ defined as

$$\overline{u}_i = \frac{1}{|\mathcal{V}_i|} \int_{\mathcal{V}_i} u \, \mathrm{d}\mathcal{V}_i \quad , \tag{6}$$

and is obtained by integration of Eq. (1) on volume element $\mathcal{V}_i$, assuming a static mesh with no deformation or movement of volume elements, which yields

$$\frac{\mathrm{d}\overline{u}_i}{\mathrm{d}t} + \frac{1}{|\mathcal{V}_i|} \sum_{\ell \in \mathcal{L}_i} \int_{\mathcal{S}_\ell} \mathcal{F}(u) \cdot \boldsymbol{n}_{i,\ell} \, \mathrm{d}\mathcal{S} = 0 \tag{7}$$

where $\boldsymbol{n}_{i,\ell}$ is the unitary outward-facing face normal of face $\mathcal{S}_\ell$ with respect to cell $\mathcal{V}_i$ computed as in [21]. In this work, only planar faces are considered, which implies that face normals are constant on each face.

Let us define the face average operator $\widetilde{\square}_\ell$ on a given face $\mathcal{S}_\ell$ as

$$\widetilde{\square}_\ell = \frac{1}{|\mathcal{S}_\ell|} \int_{\mathcal{S}_\ell} \square(\boldsymbol{x}) \, \mathrm{d}\mathcal{S} \quad . \tag{8}$$

Face average normal fluxes of Eq. (7) are also computed as

$$\widetilde{\mathcal{F}}_{i,\ell} = \frac{1}{|\mathcal{S}_\ell|} \int_{\mathcal{S}_\ell} \mathcal{F}(u) \cdot \boldsymbol{n}_{i,\ell} \, \mathrm{d}\mathcal{S} \quad . \tag{9}$$

For the sake of brevity, face average normal fluxes will also be referred as face average fluxes in this work. Eq. (7) is then transformed into the following discrete system by spatial discretization of the fluxes into the semi-discrete system

$$\frac{\mathrm{d}\overline{u}_i^\delta}{\mathrm{d}t} = -\frac{1}{|\mathcal{V}_i|} \sum_{\ell \in \mathcal{L}_i} \widetilde{\mathcal{F}}_{i,\ell}^\delta |\mathcal{S}_\ell| \quad , \tag{10}$$

where the super-index δ symbolizes a numerical approximation. It can then be clearly observed that the numerical errors in the FV formulation are introduced through the discretization of the face average fluxes and the time integrator.

A review of the most common approaches for the reconstruction of high-order face average fluxes (higher than second order) is carried out in Section 2.3. In this study, the time scheme used for time integration of Eq. (10) is a third order Runge-Kutta (RK) propagator method (RK33) [22].

2.3. Finite volume face average flux reconstruction methods

In this work, high-order discretization of the face average fluxes (higher than second order) will be used to discretize the FV equations. In high-order FV methods, the most common techniques for the numerical computation of the flux face integrals in Eq. (9) consist of quadrature methods using local values of the conserved variable at the given quadrature points (see [2,11,23,3,8,4,9])

$$\widetilde{\mathcal{F}}_{i,\ell} \approx \widetilde{\mathcal{F}}_{i,\ell}^\delta = \sum_{g \in \mathcal{G}_\ell} w_g \mathcal{F}\left(u\left(\boldsymbol{x}_g\right), \nabla u\left(\boldsymbol{x}_g\right)\right) \cdot \boldsymbol{n}_{i,g} \quad , \tag{11}$$

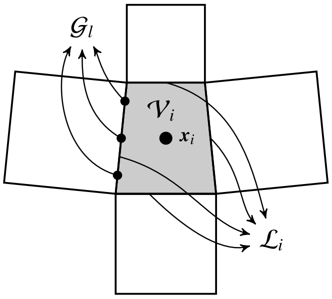


Fig. 1. Sketch of a two-dimensional mesh with quadrilateral partitioning and the different sets used to identify faces of a given cell and quadrature points of a given face.

where $\mathcal{G}_\ell$ is the set of indices of the weights w_g associated with the quadrature points $\boldsymbol{x}_g$ of face $\mathcal{S}_\ell$ (see Fig. 1). In these methods, a polynomial representation of the conserved variables in each cell is mandatory to obtain a high-order approximation of the values of the conserved variables at quadrature points of each cell. To abbreviate the formulation, the values of the conserved variables and their gradients at a given quadrature point with index g are referred to u_g and ∇u_g respectively. Eq. (11) can be rewritten in terms of the numerical values of the conserved variables at the quadrature points as

$$\widetilde{\mathcal{F}}^\delta_{i,\ell} = \sum_{g\in\mathcal{G}_\ell} w_g \mathcal{F}^\delta \left(u^{\delta,-}_g, \nabla u^{\delta,-}_g, u^{\delta,+}_g, \nabla u^{\delta,+}_g, \boldsymbol{n}_g\right) = \sum_{g\in\mathcal{G}_\ell} w_g \left(\mathcal{F}^\delta_{\mathrm{c},i,g} + \mathcal{F}^\delta_{\mathrm{v},i,g}\right) \quad , \tag{12}$$

where the numerical convective and viscous flux operators $\mathcal{F}^\delta_{\mathrm{c},i,g}$ and $\mathcal{F}^\delta_{\mathrm{v},i,g}$ have been introduced. The symbols ± represent the two possible numerical reconstructions (commonly referred to as left and right states) of the conserved variables at a given quadrature point of a non-boundary face. The left and right reconstructions, $u^{\delta,-}_g$ and $u^{\delta,+}_g$, are respectively computed using the stencils of the owner cell of the face $\mathcal{V}_{\ell,\mathrm{o}}$ and $\mathcal{V}_{\ell,\mathrm{n}}$ of face $\mathcal{S}_\ell$. Meanwhile, the right reconstruction $u^{\delta,+}_g$ is computed using the stencil of the neighbor cell of the face $\mathcal{V}_{\ell,\mathrm{n}}$. The face normal at each quadrature point $\boldsymbol{n}_g$ of face $\mathcal{S}_\ell$ with respect to their owner and neighbor cells is defined such that $\boldsymbol{n}_g \cdot (\boldsymbol{x}_\mathrm{n} - \boldsymbol{x}_\mathrm{o}) \geq 0$. The reader may refer to Section 3 for a detailed explanation of the reconstruction of the conserved variables at the quadrature points.

Remark 1. We are aware that there exist other high-order finite-volume flux reconstruction methods for structured and unstructured meshes which were initially introduced for structured grids by [12]. These methods rely on the computation of the Taylor expansion of the numerical face average flux. A 6th-order FV WENO formulation was recently proposed in [13]. An extension of the method for unstructured grids was developed in [14].

Processing of face average convective fluxes

To compute the numerical convective flux at each internal quadrature point $\mathcal{F}^\delta_{\mathrm{c},i,g}$, the following weighted expression is used

$$\mathcal{F}^\delta_{\mathrm{c},i,g} = \left[f_g \mathcal{F}_{\mathrm{c,upwind}}\left(u^{\delta,-}_g, u^{\delta,+}_g, \boldsymbol{n}_g\right) + (1-f_g)\mathcal{F}_{\mathrm{c,central}}\left(u^{\delta,-}_g, u^{\delta,+}_g, \boldsymbol{n}_g\right)\right] \mathrm{sgn}(\boldsymbol{n}_{i,g}\cdot\boldsymbol{n}_g)) \quad , \tag{13}$$

where $f_g \in [0, 1]$ is a weighting factor, $\mathcal{F}_{\mathrm{c,upwind}}$ is an upwind flux whose formulation is usually especially tailored for a given non-linear problem using Riemann solvers. For the Euler equations (Section 2.5.2) one can find Riemann solvers such as Roe [24], HLL or HLLC [25]. $\mathcal{F}_{\mathrm{c,central}}$ is a central flux which is computed in this work as

$$\mathcal{F}_{\mathrm{c,central}}\left(u^{\delta,-}_g, u^{\delta,+}_g, \boldsymbol{n}_g\right) = \frac{1}{2}\left(\boldsymbol{\mathcal{F}}_c\left(u^{\delta,-}_g\right) + \boldsymbol{\mathcal{F}}_c\left(u^{\delta,+}_g\right)\right)\cdot\boldsymbol{n}_g \quad . \tag{14}$$

For $f = f_g = 1$ upwind flux reconstruction schemes are obtained while the case $f = f_g = 0$ is representative of central schemes. In this work, both upwind and central schemes will be studied and they will be denoted with the initial letter U or C respectively.

The order of accuracy of these face average flux reconstruction methods is determined by two parameters: the order of accuracy of the interpolation method of the conserved variable at the quadrature points (analyzed in our studies in Section 4.2) and that of quadrature method used to evaluate the face average flux (discussed in Section 3.3).

Processing of face average viscous fluxes

In contrast with the convective terms, the numerical viscous fluxes involve reconstruction of gradients at the quadrature points on each side of the face. In this work, the numerical viscous flux $\mathcal{F}_{\mathrm{v},i,g}$ at each quadrature point is directly computed as

$$\mathcal{F}^{\delta}_{\mathrm{v},i,g} = \mathcal{F}_v\left(u^{\delta}_g, \nabla u^{\delta}_g\right) \cdot \boldsymbol{n}_{i,g} \quad , \tag{15}$$

where u^{δ}_g and ∇u^{δ}_g are computed with a central scheme as

$$u^{\delta}_g = \frac{1}{2}\left(u^{\delta,-}_g + u^{\delta,+}_g\right) \quad \text{and} \quad \nabla u^{\delta}_g = \frac{1}{2}\left(\nabla u^{\delta,-}_g + \nabla u^{\delta,+}_g\right) \quad . \tag{16}$$

One could also consider other types of numerical viscous fluxes such as the Interior-Penalty (IP) method or the Rebay-Bassi (RB) approach. The interested reader is referred to [26,27] for more information on numerical viscous fluxes formulations. At last, Section 3.4 discusses the computation of the gradient of the conserved variables at quadrature points, using the numerical schemes of choice of this work.

2.4. Finite volume variables reconstruction methods

This section is devoted to the description of the state of the art of the FV numerical schemes which provide high-order reconstructions of the conserved variables at the face quadrature points.

k-exact schemes for structured meshes

k-exact methods for structured meshes have been developed in [8,9]. They allow to compute high-order face average flux reconstructions using Eq. (12). They are based on the use of tensor-product of one-dimensional polynomials to represent a modal basis of the solution. The extension of this method for non-uniform structured grids was proposed in [15] and will be analyzed in this work in Section 3.2.

k-exact schemes for arbitrary meshes

There exists a variety of k-exact methods which are applicable to any kind of mesh and mesh elements. These methods are based on the computation of a p-degree polynomial to approximate the conserved variables in each cell. The polynomial approximation enables a high-order interpolation of the conserved variable at quadrature points. These interpolations are used to compute numerical fluxes using quadrature integrals Eq. (12). Several types of k-exact reconstruction methods can be listed:

1. The compact methods of [28,10] in which coefficients of the polynomial are computed iteratively using data from the closest neighbors via successive least-squares problems.
2. The non-compact method of [14], where the coefficients of a polynomial approximation of the conserved variables are approximated through several LSQ problems on each layer of neighbors.
3. The non-conservative moving least-squares (MLS) method proposed by [29] which can also be used in mesh-free formulations.
4. The non-compact least-squares (LSQ) method proposed by [6,2,11,3] which is one pf the main topic of research of this work. In these methods the conserved variables are spatially estimated by a polynomial of a given degree whose coefficients are obtained from neighbor cells through the solution of a single least-squares problem in a non-compact manner. These methods are commonly utilized in conjunction with ENO or WENO [4] methods by reconstructing a plethora of polynomial approximations of the conserved variables at each cell using different stencil topologies. These polynomials can then be combined by measuring their smoothness ensuring the correct treatment of solution discontinuities.

2.5. Choice of conservative equations

Three systems of conservation laws are considered in this study: the linear advection equation, the Euler equations and the Navier-Stokes equations.

2.5.1. Linear advection equation

The linear advection equation is a single equation conservation law in which the flux operator $\mathcal{F}$ of the conservation law is linear $\mathcal{F} = \boldsymbol{c}u$, where $\boldsymbol{c}$ is the linear advection velocity and it is supposed to be constant in this work. In this case, Eq. (1) simplifies to

$$\frac{\partial u}{\partial t} + \boldsymbol{c} \cdot \nabla u = 0 \quad . \tag{17}$$

Supposing planar faces, the finite volume formulation of Eq. (17) can be written as

$$\frac{\mathrm{d}\overline{u}_i}{\mathrm{d}t} + \frac{1}{|\mathcal{V}_i|} \sum_{\ell \in \mathcal{L}_i} \widetilde{u}_{\ell} \boldsymbol{c} \cdot \widetilde{\boldsymbol{n}}_{i,l} |\mathcal{S}_{\ell}| = 0 \quad . \tag{18}$$

In future references to the linear advection equation in two-dimensional context, the linear advection velocity will be expressed as

$$\boldsymbol{c} = c(\cos\zeta, \sin\zeta) \tag{19}$$

where ζ is the advection velocity angle.

2.5.2. Euler equations

The Euler equations are written in conservative form as

$$\frac{\partial \boldsymbol{U}}{\partial t} + \nabla \cdot \mathcal{F}(\boldsymbol{U}) = 0 \quad , \tag{20}$$

where $\boldsymbol{U}$ is the vector of conservative variables given by

$$\boldsymbol{U} = \begin{pmatrix} \rho \\ \rho\boldsymbol{u} \\ \rho E \end{pmatrix} \quad , \tag{21}$$

ρ is the fluid density, $\rho\boldsymbol{u}$ the momentum, $\boldsymbol{u}$ the velocity and ρE is the total energy. The flux operator of the Euler equations can be written as

$$\mathcal{F}(\boldsymbol{U}) = \begin{pmatrix} \rho\boldsymbol{u} \\ \rho\boldsymbol{u}\otimes\boldsymbol{u} + PI_{N_d} \\ (\rho E + P)\boldsymbol{u} \end{pmatrix} \quad , \tag{22}$$

where the symbol $\otimes$ represents the dyadic operator, P is the pressure and $I_{N_d} \in \mathbb{R}^{N_d \times N_d}$ is the identity matrix. The non-linear system is closed with the equation of state

$$\rho E = \frac{P}{\gamma - 1} + \frac{1}{2}\rho\boldsymbol{u}\cdot\boldsymbol{u} \quad , \tag{23}$$

where γ is the ratio of specific heats. The primitive variables of the Euler equations ρ, $\boldsymbol{u}$, P and the temperature Θ can be computed from the conservative variables using the equation of state Eq. (23) and the following additional expressions:

$$\boldsymbol{u} = \frac{\rho\boldsymbol{u}}{\rho} \;,\; P = (\gamma - 1)\left(\rho E - \frac{1}{2}\rho\boldsymbol{u}\cdot\boldsymbol{u}\right) \;,\; \Theta = \frac{P}{\rho r} \;,\; c_p = \frac{\gamma r}{(\gamma - 1)} \quad , \tag{24}$$

where r is the gas constant and c_p is the heat capacity at constant pressure. The flux operator of the Euler equations is non-linear and therefore the only possibility of obtaining a formally high-order finite volume numerical solution is by using the high-order face average flux reconstructions such as Eq. (12).

2.5.3. Compressible Navier-Stokes equations

The compressible Navier-Stokes equations are a system of non-linear equations similar to the Euler equations. Both sets of equations share the same conserved (Eq. (21)) and primitive variables (Eq. (24)) and have similar flux operators. However, the Navier-Stokes equations present additional viscous terms which depend on the values and the gradients of the conserved and/or the primitive variables. These viscous terms are denoted as $\mathcal{F}_v(\boldsymbol{U}, \nabla\boldsymbol{U})$. The flux operator of the Navier-Stokes equations can be written as

$$\mathcal{F}(\boldsymbol{U}) = \begin{pmatrix} \rho\boldsymbol{u} \\ \rho\boldsymbol{u}\otimes\boldsymbol{u} + PI_{N_d} \\ (\rho E + P)\boldsymbol{u} \end{pmatrix} + \begin{pmatrix} 0 \\ -\boldsymbol{\sigma} \\ -\frac{\mu c_p}{Pr}\nabla\Theta - \boldsymbol{\sigma}\cdot\boldsymbol{u} \end{pmatrix} \quad , \tag{25}$$

where μ is the kinematic viscosity, Pr is the Prandtl number and $\boldsymbol{\sigma}$ is the viscous stress tensor defined as

$$\boldsymbol{\sigma} = \mu\left(\nabla\boldsymbol{u} + (\nabla\boldsymbol{u})^T - \frac{2}{3}\nabla\cdot\boldsymbol{u}I_{N_d}\right) \quad . \tag{26}$$

The velocity and temperature gradients can be computed from the gradients of the conservative variables through the chain rule and the relationship between the conservative and primitive variables expressed in Eq. (24).

3. k-exact finite volume schemes

This section addresses the different non-compact k-exact schemes discussed in this work. Section 3.1 discusses least-squares k-exact schemes compatible with both structured and unstructured meshes. Section 3.2 discusses fully constrained k-exact schemes only compatible with structured meshes. The formulations are written supposing two-dimensional problems. The extension to one-dimensional and three-dimensional problems is trivial. However, remarks regarding the extension to different number of dimensions will be added when needed.

3.1. k-exact least-squares schemes

The present section addresses the least-squares k-exact reconstruction approach in unstructured meshes used in this work. The methodology has been developed by [2] and [5] whose work is a continuation of [6] and [7]. Firstly, let us consider a reference coordinate space for each cell $\mathcal{V}_i$ denoted as $\boldsymbol{\xi} = (\xi, \eta)$ related to the global coordinates $\boldsymbol{x} = (x, y)$ via a linear mapping:

$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} x_0 \\ y_0 \end{pmatrix} + \mathcal{J} \cdot \begin{pmatrix} \xi \\ \eta \end{pmatrix} \quad , \tag{27}$$

where (x_0, y_0) are the coordinates of an arbitrary reference node of the cell and $\mathcal{J}$ is the Jacobian matrix computed as

$$\mathcal{J} = \begin{pmatrix} x_1 - x_0 & x_2 - x_0 \\ y_1 - y_0 & y_2 - y_0 \end{pmatrix} \quad . \tag{28}$$

In the latter expression, sub-indices 1 and 2 refer to two nodes of cell i which are connected to the chosen reference node. The choice of reference node is arbitrary and therefore, the properties of these schemes are dependent on the vertex-cell connectivity of the mesh. It is important to note that any integral evaluation in the global coordinate space can be transformed into the reference space as

$$\frac{1}{|\mathcal{V}_i|} \int\limits_{\mathcal{V}_i} T(\boldsymbol{x})\, \mathrm{d}\mathcal{V} = \frac{1}{|\mathcal{V}_i|} \int\limits_{\mathcal{V}_i} T(\boldsymbol{x})\, \mathrm{d}x\, \mathrm{d}y = \frac{1}{|\mathcal{V}'_i|} \int\limits_{\mathcal{V}'_i} T(\boldsymbol{\xi})\, \mathrm{d}\xi\, \mathrm{d}\eta \quad , \tag{29}$$

where $|\mathcal{V}'_i| = |\mathcal{V}_i||\mathcal{J}^{-1}|$ is the volume measure of cell i in the reference space and $|\mathcal{J}^{-1}|$ is the determinant of the inverse Jacobian matrix. In these methods a polynomial $\mathcal{P}_i(\boldsymbol{\xi})$ is built for each cell i of the domain as

$$\mathcal{P}_i(\boldsymbol{\xi}) = \sum_{\gamma=1}^{K} a_\gamma \phi_\gamma(\boldsymbol{\xi}) = u_i^\delta(\boldsymbol{\xi}) \quad , \tag{30}$$

where the superscript δ denotes a numerical approximation of a given function over a cell and the functions $\{\phi_\gamma\}$ are a family of polynomial functions of the form

$$\forall \gamma \in [1, K] \quad \phi_\gamma(\boldsymbol{\xi}) = \Psi_\gamma(\boldsymbol{\xi}) - C_\gamma \quad , \tag{31}$$

such that

$$\{\Psi_\gamma(\boldsymbol{\xi}), \gamma \in [1, K]\} = \{(\xi - \xi_i)^q (\eta - \eta_i)^r, q \geq 0, r \geq 0, q + r \leq p\} \quad . \tag{32}$$

In Eqs. (31) and (32), (ξ_i, η_i) is the centroid of cell i in the reference space, p is the degree of the polynomial, K is the cardinality of the bases $\{\phi_\gamma(\boldsymbol{\xi})\}$ and $\{\Psi_\gamma(\boldsymbol{\xi})\}$ and depends on the spatial dimension N_d as

$$K = \binom{p + N_d}{p} \quad , \tag{33}$$

and $\{C_\gamma\}$ is a family of constants whose values will be determined further into this section.

In general, the coefficients a_γ of Eq. (30) are determined by setting integral equations for the root cell i and a number of its closest neighbors, which together with the root cell form the stencil named C_i with cardinality $M = |C_i|$. A discussion on the definition of the stencil for these LSQ methods is developed in Section 3.1.4. In the least-squares schemes studied in this section the number of neighbors is chosen such that $M > K$. The setting $M = K$ is specific to a small family of k-exact schemes for structured meshes as discussed in Section 3.2.

3.1.1. Definition of the LSQ problem

The LSQ problem is generated by equating the average of the reconstructed polynomial $\mathcal{P}_i$ on each cell $\mathcal{V}'_j$ of the stencil C_i with the cell average $\overline{u}_j$ of the solution as in

$$\forall\, j \in C_i \setminus \{i\} \quad \frac{1}{|\mathcal{V}'_j|} \int\limits_{\mathcal{V}'_j} \mathcal{P}_i(\boldsymbol{\xi})\, \mathrm{d}\xi\, \mathrm{d}\eta = \overline{u}_j \quad , \tag{34}$$

which can be rearranged as

$$\forall\, j \in C_i \setminus \{i\} \quad \sum_{\gamma=1}^{K} a_\gamma \left(\frac{1}{|\mathcal{V}'_j|} \int\limits_{\mathcal{V}_j} \phi_\gamma(\boldsymbol{\xi})\, \mathrm{d}\xi\, \mathrm{d}\eta \right) = \overline{u}_j \quad . \tag{35}$$

The least-squares problem is constrained on cell i by the equation

$$\frac{1}{|\mathcal{V}_i'|}\int\limits_{\mathcal{V}_i'} \mathcal{P}_i(\boldsymbol{\xi})\,\mathrm{d}\xi\,\mathrm{d}\eta = \overline{u}_i \quad , \tag{36}$$

ensuring that the cell average is exactly reconstructed by the polynomial $\mathcal{P}_i$.

Let us impose that the constant polynomial obtained for $q = r = 0$ in Eq. (32) be assigned to index $\gamma = 1$ with a null constant $C_{\gamma=1} = 0$ and that the integral of ϕ_γ on cell i be zero for every other γ:

$$\forall\gamma > 1 \quad \frac{1}{|\mathcal{V}_i'|}\int\limits_{\mathcal{V}_i'} \phi_\gamma(\boldsymbol{\xi})\,\mathrm{d}\xi\,\mathrm{d}\eta = 0 \quad . \tag{37}$$

By inserting the previous assumptions in Eq. (36), the following values for the constants C_γ are obtained (as in the studies of [7,2]):

$$\forall\gamma > 1 \quad C_\gamma = \frac{1}{|\mathcal{V}_i'|}\int\limits_{\mathcal{V}_i'} \Psi_\gamma(\boldsymbol{\xi})\,\mathrm{d}\xi\,\mathrm{d}\eta \quad , \tag{38}$$

from which ensues the values of a_1, as described in [2,11]:

$$a_1 = a_1\phi_1 = a_1\Psi_1 = \overline{u}_i \quad . \tag{39}$$

We stress out that this choice is arbitrary and will indeed modify the numerical properties and stability of the schemes, although the analysis of this topic will not be addressed in this work.

From the above expressions and the constraint imposed in Eq. (36), the over-constrained problem may be recast in matrix form as

$$\mathcal{A}\mathcal{X} = \boldsymbol{b} \quad , \tag{40}$$

where $\mathcal{A}$ is a matrix in $\mathbb{R}^{(M-1)\times(K-1)}$ with elements

$$\mathcal{A}_{j\gamma} = \frac{1}{|\mathcal{V}_j'|}\int\limits_{\mathcal{V}_j'} \phi_{\gamma+1}(\boldsymbol{\xi})\,\mathrm{d}\xi\,\mathrm{d}\eta = \frac{1}{|\mathcal{V}_j'|}\int\limits_{\mathcal{V}_j'} \Psi_{\gamma+1}(\boldsymbol{\xi})\,\mathrm{d}\xi\,\mathrm{d}\eta - C_{\gamma+1} \quad . \tag{41}$$

Vectors $\mathcal{X}$ and $\boldsymbol{b}$ belong to $\mathbb{R}^{K-1}$ and $\mathbb{R}^{M-1}$ respectively and are set as

$$\forall\gamma \in [1, K-1] \quad \mathcal{X}_\gamma = a_{\gamma+1} \quad , \tag{42}$$

and

$$\forall j \in C_i \setminus \{i\} \quad \boldsymbol{b}_j = \overline{u}_j - \overline{u}_i \quad . \tag{43}$$

Remark 2. In this work, all terms in matrix $\mathcal{A}$ are computed using an exact numerical integration quadrature of the polynomial basis up to machine precision. In one-dimensional meshes, this is performed using Gauss-Legendre quadratures of appropriate order. In two-dimensional and three-dimensional meshes, the volume integrals are evaluated by decomposing the tensor-product elements (quadrilaterals or hexahedrons) in N_d-simplex elements (triangles or tetrahedrons) and evaluating the volume integrals on those elements. To do so, this work relies on the quadratures schemes described in [30] for triangles (2-simplexes) and the quadratures scheme of [31] for tetrahedrons (3-simplexes). The quadrature points positions and weights of these schemes were obtained using the QUADPY software [32]. The decomposition of tensor-product elements in simplex cells does not incur geometric discretization errors if one only considers tensor-product elements with planar faces. Therefore, the volume integrals are evaluated by subdividing the cells into a set of N_d-simplexes and applying quadrature rules on those subelements. As the determinant of the Jacobian of the isoparametric transform of simplex elements is constant, the integrals can be exactly evaluated up to machine round-off precision with an appropriate quadrature. It is important to note that the exact evaluation of volume integrals in arbitrary tensor-product elements with Gauss-Legendre quadratures is also possible but one needs to take into account that the determinant of the isoparametric Jacobian transformation (needed to evaluate Gauss-Legendre quadratures) is a 3rd degree polynomial [33] in the worst case scenario (hexahedron cells). Therefore, the Gauss-Legendre quadrature should be capable of exactly integrating a $p+3$ degree polynomial.

3.1.2. Resolution of the least-squares problem

The over-constrained problem described in Eq. (40) is solved with the least-squares L_2-minimization approach, which implies the computation of the pseudoinverse $\mathcal{A}^+$ of the matrix $\mathcal{A}$. In this work, the pseudoinverse is obtained using the SVD (Singular Value Decomposition) method [34,5], implemented in the EIGEN library [35]. This process is only performed during the simulation's initialization process, because $\mathcal{A}^+$ uniquely depends on the geometrical properties of the mesh. The solution vector can be then obtained as

$$\mathcal{X} = \mathcal{A}^+ \boldsymbol{b} \quad . \tag{44}$$

The choice of stencil and the polynomial order will determine the properties of the least-squares spatial interpolation schemes. This issue will be discussed in Section 3.1.4.

3.1.3. Distance weighting in the LSQ problem

Several studies [6,36,17] have pointed out the influence of the addition of distance weighting methods to the least-squares problem on the numerical properties of the k-exact schemes. The distance-weighted least-squares problem is obtained by rewriting the previous least-squares problem as

$$\mathcal{A}^w \mathcal{X} = \boldsymbol{b}^w \quad , \tag{45}$$

with

$$\mathcal{A}^w_{j\gamma} = \frac{1}{||\boldsymbol{x}_j - \boldsymbol{x}_i||^\alpha} \mathcal{A}_{j\gamma} \quad , \tag{46}$$

and

$$\boldsymbol{b}^w_j = \frac{1}{||\boldsymbol{x}_j - \boldsymbol{x}_i||^\alpha} \boldsymbol{b}_j \quad , \tag{47}$$

where, α is the weighting exponent, and $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ are the vector coordinates of the centers of cells $\mathcal{V}_i$ and $\mathcal{V}_j$. The solution vector can then be computed as

$$\mathcal{X} = \mathcal{A}^{w,+} \boldsymbol{b}^w \quad , \tag{48}$$

where $\mathcal{A}^{w,+}$ is the pseudoinverse matrix of $\mathcal{A}^w$. Schemes obtained with distance weighting least-squares method will be designated as LSQpW. In this work the weighting exponent is fixed as $\alpha = \frac{5}{2}$, chosen as a compromise between the scheme accuracy and the conditioning of matrix $\mathcal{A}$. Constraint equation (36) and constants C_γ from Eq. (38) remain unchanged with respect to the non-weighted least-squares problem.

3.1.4. Definition of the k-exact LSQ schemes

The accuracy of polynomial $\mathcal{P}_i$ used for the reconstruction of the conservative variable greatly depends on the choice of stencil C in general. The choice of stencil was proven crucial for the accuracy of k-exact schemes in unstructured meshes [3]. Nonetheless, few studies have analyzed the effects of the stencil shape and topology on the accuracy of these schemes for smooth meshes using quadrilateral or hexahedral elements. Most studies use stencils by gathering a given number of topological neighbors with connected faces or vertices with each other. The number of neighbor elements in the stencil is commonly computed as a function of the polynomial cardinal functions using the β coefficient as

$$\beta = \frac{M-1}{K-1} \quad . \tag{49}$$

Please note that $M-1$ is the number of neighbors of a given root cell used to reconstruct a least-squares polynomial and $K-1$ is the size of the polynomial basis for the least-squares procedure given that the constant polynomial was specifically constrained as stated in Eq. (39).

In practice, the value $\beta = 2$ is usually adopted. However, this condition generates ill-formed stencils in smooth meshes with quadrilateral and hexahedral meshes, because it is not possible to ensure topologically symmetric stencils for every cell if the value of β is fixed to a value which is not representative of the number of topological neighbors of a given cell. If the connectivity of the stencil of neighboring cells is dissimilar, the truncation error will not behave regularly, therefore some error canceling properties of the truncation error will be lost even in smoothly varying grids.

In this study, it has been decided to associate a number of neighbors M and a parameter β to a given topological shape of the neighbor cells of the root cell which share a given face with it (face neighbor cells) similarly to the methods described in [6,17,18]. This ensures geometric symmetry of the stencils in structured meshes and a smooth variation of the truncation error, provided that the mesh is also locally smooth. The choice to generate the reconstruction stencil using cell face neighbors is motivated by the will to ensure an almost constant value of β.

Geometric symmetry of the stencil is ensured by associating a number of neighbors equal to the actual number of neighbors of a face. Such a property provides a smooth variation of the truncation error. Therefore, it is possible to generate

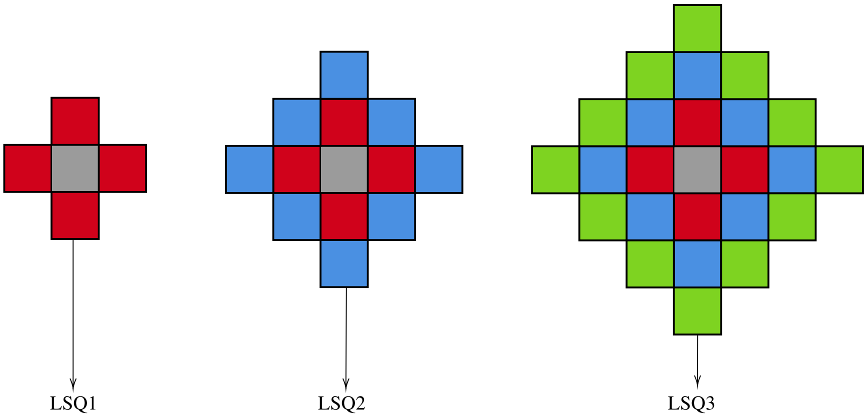


Fig. 2. Two-dimensional face neighbors stencils of the LSQ1-2-3 schemes.

Table 1
Number of neighbors M of the different LSQ schemes in uniform and structured meshes as a function of the number of dimensions of the problem.

	LSQ1	LSQ2	LSQ3	LSQ4
1D	3	5	7	9
2D	5	13	25	41
3D	7	25	63	129

Table 2
β parameter of the different LSQ schemes in uniform and structured meshes as a function of the number dimensions of the problem.

	LSQ1	LSQ2	LSQ3	LSQ4
1D	2	2	2	2
2D	2	12/5	8/3	20/7
3D	2	8/3	62/19	128/34

what we call face neighbor layers with the following algorithm. First, for each root cell control volume $\mathcal{V}_i$, all face neighbors are added to the stencil C_i generating the first face neighbor layer. If further neighbor layers are needed, the previous process is repeated for each cell added in the previous iteration (ensuring unique indices in C_i) until the stencil contains a number of face neighbor layers equal to the polynomial order p of the scheme.

Regarding the LSQ schemes that will be analyzed in this work, we will restrict the study to those with polynomial degree such that $p = 1, 2, 3$ and 4. The resulting schemes are denoted as LSQp, with $p = 1, 2, 3$ and 4. Moreover, the effect of the weighting method in the least-squares problem resolution will be studied for $p = 3$. The subsequent weighted scheme will be referred as LSQ3W.

Least-squares stencils and face neighbor layers for schemes LSQp with $p = 1, 2$ and 3 are represented for two-dimensional problems in Fig. 2. These figures allow to identify the stencil's geometry of some of these schemes in two-dimensional meshes.

To better understand the value of the parameter β with the different combinations of polynomial order p and number of dimensions N_d, Table 1 shows the number of neighbors of a given stencil for the different LSQ schemes and Table 2 represents the value of the β parameter as a function of the dimensions of the problem and the different LSQ schemes. In these tables, it is possible to observe that the parameter β increases with the number of dimensions of the problem for a given polynomial order. It can even reach values higher than $\beta > 3$ in some three-dimensional cases. With such high values of β, it could be interesting to reduce by one the neighbor layer index used to build the k-exact scheme. However, to remain consistent for all spatial dimensions, we decided not to change the index of the neighbor layer used to build the schemes, independently of the number of dimensions. It is also worth noting that reducing by one the index of the neighbor layer yields always $\beta < 2$ in the cases presented in Table 2.

3.2. k-exact schemes for structured meshes of tensor-product elements

Recent developments made by [15] allow to obtain multidimensional k-exact schemes using a fully constrained problem for structured non-uniform meshes made up of tensor elements (quadrilaterals and hexahedrons). To develop these methods, a structured mesh with principal directions $x-y$ will be supposed. The extension to one and three-dimensional problems is straightforward. These schemes use the following linear mapping

$$\begin{pmatrix} x \\ y \end{pmatrix} = \begin{pmatrix} x_i \\ y_i \end{pmatrix} + \mathcal{J} \cdot \begin{pmatrix} \xi \\ \eta \end{pmatrix} \quad , \tag{50}$$

where (x_i, y_i) are the coordinates of the root cell centroid and

$$\mathcal{J} = \begin{pmatrix} h_i & 0 \\ 0 & h_i \end{pmatrix} \quad . \tag{51}$$

The polynomial basis used by the schemes belongs to the polynomial basis of tensor product one dimensional polynomials of degree p

$$\{\Phi(\boldsymbol{\xi})\} = \left\{ \bigotimes_{i=1}^{N_d} (1, \xi_i^1, \ldots, \xi_i^p) \right\} \quad . \tag{52}$$

Please note that the 0th degree of freedom is included polynomial basis, opposed to the choice made in k-exact least-squares schemes where this polynomial basis is supposed equal to the cell average (see Section 3.1). Moreover, the maximum monomial degree is $N_d p$. The cardinal of the polynomial basis is redefined for tensor product polynomials as

$$K = (p+1)^{N_d} \quad . \tag{53}$$

Regarding the neighbors used to obtain the polynomial coefficients, as the 0th degree polynomial basis is unknown, the root cell i will also be added to the central stencil C_i. These k-exact schemes build their stencils using cell vertex neighbors instead of cell face neighbors. For the sake of brevity, the discussion of the generation of the vertex neighbors based stencils is avoided as it can be straightforwardly understood from the generation face neighbors based stencils discussed in Section 3.1.4. The number of neighbors $M = |C_i|$ for a given cell vertex neighbor layer index N_v in structured meshes is

$$M = (2N_v + 1)^{N_d} \quad . \tag{54}$$

In the latter expression it is possible to observe that the condition $K = M$ (which is needed to ensure a fully-constraint problem) imposes the following condition for even values of p

$$N_v = \frac{p}{2} \quad . \tag{55}$$

Therefore, a given tensor product polynomial of even degree p can be computed without using least-squares methods in a structured mesh by using the correct number of vertex neighbor layers. Odd degree polynomials can also be considered, although one needs to consider a different family of polynomials and face-based stencils as explained in [15]. Therefore, we particularize our study for even degree polynomials. In these cases, the least-squares problem defined in Section 3.1 can be rewritten as a linear system as $M = K$. For each neighbor cell in the C_i the following constraint equation is used to build the matrix $\mathcal{A} \in \mathbb{R}^{K \times K}$ as

$$\forall\, j \in C_i \quad \sum_{\gamma=1}^{K} \frac{a_\gamma}{|\mathcal{V}'_j|} \int\limits_{\mathcal{V}_j} \phi_\gamma(\boldsymbol{\xi})\, \mathrm{d}\xi\, \mathrm{d}\eta = \overline{u}_j \quad . \tag{56}$$

The expression of the terms of matrix $\mathcal{A}$ is

$$\mathcal{A}_{nm} = \frac{1}{|\mathcal{V}'_n|} \int\limits_{\mathcal{V}_n} \phi_m(\boldsymbol{\xi})\, \mathrm{d}\xi\, \mathrm{d}\eta \quad . \tag{57}$$

Remark 3. In this work, all the terms from matrix $\mathcal{A}$ are computed using a numerical integration quadrature which integrates exactly all the polynomial basis up to machine precision. Compared to the LSQ schemes, these methods require higher order numerical quadrature if one wants to integrate exactly all the polynomial basis. For example, the maximum polynomial degree is pN_d, which has a value of 12 for a 4th degree polynomial.

Meanwhile, the right-hand-side vector $\boldsymbol{b} \in \mathbb{R}^M$ is equal to the cell averages of all the cells present in C_i. The polynomial coefficients $\mathcal{X} \in \mathbb{R}^K$ can then be obtained by building the inverse matrix $\mathcal{A}^{-1}$ and using the previous definition of the right-hand-side vector:

$$\mathcal{X} = \mathcal{A}^{-1}\boldsymbol{b} \quad . \tag{58}$$

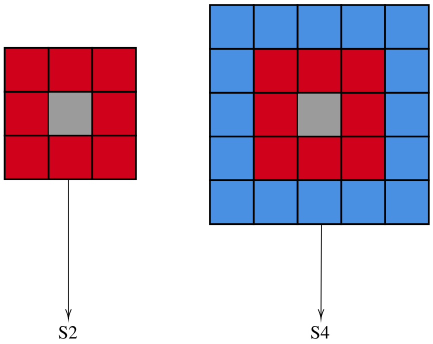


Fig. 3. Two-dimensional vertex neighbors stencils of the S2 and S4 schemes.

Table 3
Number of neighbors M of the different Q schemes in uniform and structured meshes as a function of the number of dimensions of the problem.

	S2	S4
1D	3	5
2D	9	27
3D	25	125

In uniform structured meshes these schemes are equivalent to the finite-volume k-exact schemes for uniform structured meshes studied in [8]. When used in the discretization of linear equations, the schemes presented in this section are equivalent to those described in [37].

It is important to note that these schemes have only been developed for uniform and non-uniform structured meshes and therefore, they will only be used for comparison purposes in those cases.

Remark 4. The extension of the S schemes to quadrilateral or hexahedral arbitrarily-oriented structured meshes or unstructured grids is not straightforward due to the difficulty of choosing a suited reference frame to compute the polynomial. In our studies, we have observed that the numerical properties such as dissipation, dispersion and stability of the S schemes are heavily dependent on the reference frame chosen to generate the transformation to the reference space. Moreover, quadrilateral or hexahedral unstructured grids might have different numbers of vertex neighbors for a given layer index than structured meshes. This implies that one needs to reformulate the polynomial coefficients computation problem in these cases. Several choices can be made to deal with those issues such as recasting the fully constrained problem into a least-squares problem by modifying the polynomial basis, adding more cell face/vertex neighbors, etc. These choices will be analyzed in future studies.

3.2.1. *Definition of the S schemes*

In this study, the k-exact interpolation schemes described in Section 3.2 will be denoted with the S symbol. We particularize our study for S schemes with tensor-product polynomial degrees $p = 2$ and 4. The resulting schemes are denoted as S2 and S4 respectively.

Fig. 3 represents the stencil and neighbor cells used to build the interpolation operators of schemes S2 and S4.

Table 3 shows the number of neighbors M used by S2 and S4 as a function of the number of spatial dimensions of the problem. This table can be compared with the number of neighbors of LSQ schemes showed in Table 1, to assess the equivalence in terms of number of neighbors (which are directly related to the computational cost) between LSQ and S schemes. The S2 scheme uses same number of neighbors than the LSQ1 scheme in one-dimensional problems, although for two-dimensional and three dimensional problems its number of neighbors is much closer to that of the LSQ2 scheme. Regarding the S4 scheme, it presents the same number of neighbors than the LSQ2 scheme in one dimensional problems. Meanwhile, in two-dimensional and three three-dimensional problems, the S4 scheme uses a very similar number of neighbors than the LSQ3 and LSQ4 schemes respectively.

3.3. *Quadrature points and conserved variables face averages*

This section is devoted to the analysis of the number of quadrature points needed to compute the face average fluxes in the FV formulation using tensor-product elements.

Gauss-Legendre quadratures [38,39] allow the integration in a given face with index ℓ of a tensor-product element of a tensor-product polynomial of degree p by using a quadrature such that

Table 4
Number of quadrature points used with LSQ schemes as a function of the dimensions of the problem N_d and the polynomial order p.

	$p=1$	$p=2$	$p=3$	$p=4$
1D	1	1	1	1
2D	1	2	2	3
3D	1	4	4	9

$$|\mathcal{G}_\ell| = \text{ceil}\left(\frac{p+1}{2}\right)^{N_d-1} \tag{59}$$

where $|\mathcal{G}_\ell|$ is the number of quadrature points of face ℓ. The use of Gauss-Legendre quadratures allows to numerically compute the polynomial basis integrals along the faces as

$$\frac{1}{|\mathcal{S}'_\ell|}\int\limits_{\mathcal{S}'_\ell} \phi_\gamma\left(\boldsymbol{\xi}\right) \mathrm{d}\xi\, \mathrm{d}\eta = \sum_{g\in\mathcal{G}_\ell} w_g \phi_\gamma\left(\boldsymbol{\xi}_g\right) \tag{60}$$

where $\boldsymbol{\xi}_g$ denotes the coordinates of quadrature points in the reference space and w_g are the Gauss-Legendre quadrature weights which also include the isoparametric Jacobian transform determinant of the given face. It is important to note the polynomial basis group of LSQ schemes is contained in the tensor-product polynomial basis of S schemes. Eq. (60) holds true provided that the determinant of the isoparametric transformation of the given tensor-product element face is constant in all of its points. If this property is not fulfilled (which might be the case in hexahedral elements), numerical errors will be introduced in the evaluation of Eq. (60) through Gauss-Legendre quadratures. In these cases, Eq. (60) has to be rewritten as an approximation when evaluated in an arbitrary face of a hexahedron. These numerical errors could be reduced by increasing the number of quadrature points, knowing that, in the worst case scenario, the metric terms of an arbitrary face are a 2nd degree polynomial [21] for hexahedrons.

When dealing with non-linear fluxes, the number of quadrature points is usually obtained from Eq. (59) (see [23]) and it is shown in Table 4. It is important to note that, with non-linear fluxes, this choice of quadrature points will produce under-integration as in non-linear cases the flux polynomial representation is a product of the conserved variables. This can cause additional numerical errors and/or aliasing effects as described in [40].

3.4. Gradient computation

In order to compute the high-order viscous face average fluxes as described in Section 2.3 and Eq. (15), one needs to compute the gradients of the polynomial representative of the conserved variables at each quadrature point. To do so, the gradients of the polynomial $\mathcal{P}_i$ of each conserved variables at each cell i are computed in the reference space (ξ, η) yielding $\nabla^{\delta,\pm}_{\boldsymbol{\xi}} u$. The polynomial gradient in the reference space is then extrapolated to the quadrature points of each face of the cell by transforming the quadrature points coordinates into the reference space using the Jacobian matrix $\mathcal{J}_i$ associated to each cell. Next, this gradient in the reference space is transformed to the physical reference frame following the expression from [2]

$$\nabla^{\delta,\pm} u_g = \left(\mathcal{J}^{-1}\right)^T \nabla^{\delta,\pm}_{\boldsymbol{\xi}} u_g \quad . \tag{61}$$

At last, the gradient at each quadrature point is used to build the face average viscous flux term using Eq. (15) and Eq. (16).

4. Order of accuracy

This section is devoted to the analysis of the order of accuracy of LSQ and S schemes via the analysis of the truncation error and the k-exact properties of the reconstruction polynomials.

4.1. Linear advection spatial truncation error analysis in uniform meshes

The spatial truncation error is defined as the difference between the discretized and analytical equation. Truncation error is computed in the linear advection equation (18) neglecting time integration errors and supposing planar faces with constant metrics as

$$\mathcal{E}_i = \frac{1}{|\mathcal{V}_i|}\sum_{\ell\in\mathcal{L}_i}\left(\widetilde{u}_\ell - \widetilde{u}^\delta_\ell\right)\boldsymbol{c}\cdot\widetilde{\boldsymbol{n}}_{i,\ell}|\mathcal{S}_\ell| \quad , \tag{62}$$

where the numerical face average $\widetilde{u}^\delta_\ell$ is computed as

Table 5
Leading truncation error term and order of accuracy of the different one-dimensional least-squares schemes.

	$\mathcal{E}$	Order of accuracy
ULSQ1	$-0.083ch^2 \left.\frac{\partial^3 u}{\partial x^3}\right\vert_i + O\left(h^3\right)$	2nd
CLSQ1	$-0.083ch^2 \left.\frac{\partial^3 u}{\partial x^3}\right\vert_i + O\left(h^4\right)$	2nd
ULSQ2	$0.283ch^3 \left.\frac{\partial^4 u}{\partial x^4}\right\vert_i + O\left(h^4\right)$	3rd
CLSQ2	$-0.114ch^4 \left.\frac{\partial^5 u}{\partial x^5}\right\vert_i + O\left(h^6\right)$	4th
ULSQ3	$0.058ch^4 \left.\frac{\partial^5 u}{\partial x^5}\right\vert_i + O\left(h^5\right)$	4th
CLSQ3	$0.058ch^4 \left.\frac{\partial^5 u}{\partial x^5}\right\vert_i + O\left(h^6\right)$	4th
ULSQ3W	$0.017ch^4 \left.\frac{\partial^5 u}{\partial x^5}\right\vert_i + O\left(h^5\right)$	4th
CLSQ3W	$0.017ch^4 \left.\frac{\partial^5 u}{\partial x^5}\right\vert_i + O\left(h^6\right)$	4th
ULSQ4	$0.331ch^5 \left.\frac{\partial^6 u}{\partial x^6}\right\vert_i + O\left(h^6\right)$	5th
CLSQ4	$0.139ch^6 \left.\frac{\partial^7 u}{\partial x^7}\right\vert_i + O\left(h^8\right)$	6th

$$\widetilde{u}_\ell^\delta = f\left(\frac{1+\text{sign}(\mathbf{c}\cdot\widetilde{\boldsymbol{n}}_{i,\ell})}{2}\widetilde{u}_\ell^{\delta,-} + \frac{1-\text{sign}(\mathbf{c}\cdot\widetilde{\boldsymbol{n}}_{i,\ell})}{2}\widetilde{u}_\ell^{\delta,+}\right) + (1-f)\frac{1}{2}\left(\widetilde{u}_\ell^{\delta,-} + \widetilde{u}_\ell^{\delta,+}\right) \quad . \tag{63}$$

The first term of the expression is representative of an upwind numerical flux while the second term is representative of a central numerical flux. We recall that upwind schemes (denoted with a starting U symbol) and central schemes (denoted with a starting C symbol) use $f = 1$ and $f = 0$ respectively in all quadrature points. Eq. (62) can be further simplified in uniform meshes in which the characteristic edge size h_i is constant $h_i = h$ as

$$\mathcal{E}_i = \frac{1}{h}\sum_{\ell\in\mathcal{L}_i}\left(\widetilde{u}_\ell - \widetilde{u}_\ell^\delta\right)\mathbf{c}\cdot\widetilde{\boldsymbol{n}}_{i,\ell} \quad . \tag{64}$$

The latter expression allows to obtain the order of accuracy and the leading term of the truncation error in uniform meshes by using face and cell average Taylor expansion series of the conserved variable u. In addition, numerical values $\widetilde{u}_\ell^\delta$ can be computed by integrating analytically the polynomials resulting from the LSQ reconstruction.

Table 5 shows the leading truncation error term and order of accuracy of different k-exact least-squares schemes in the one-dimensional linear advection problem using uniform structured meshes. In these results it is possible to observe that all ULSQp schemes show at least $p+1$ order of accuracy, meaning that they respect the k-exact reconstruction condition. Moreover, the CLSQp schemes also show at least $p+1$ order of accuracy. However, it is important to note that the central variants of the LSQ schemes yield $p+2$ order of accuracy if p is even. Furthermore, central schemes always present truncation error terms which have even exponents with h. These assessments are commonly observed in the finite-difference and finite-volume literature. At last, compared to the default LSQ3 scheme the weighting distance LSQ3W scheme allows to reduce the amplitude of the leading truncation error term by a factor close to three.

Regarding the comparison of one-dimensional S and LSQ schemes, the S2 scheme has the same stencil size as the LSQ1 scheme while the S4 scheme has the same stencil size as the LSQ2 scheme. Table 6 shows the leading truncation error terms and order of accuracy of these two schemes. In these results it is possible to observe that the order of accuracy of the S2 scheme variants is similar to that of the LSQ2 scheme variants. Nonetheless, the latter has a bigger stencil and its leading truncation error term is bigger. Moreover, the S4 scheme has the same order of accuracy than the LSQ4 schemes although the LSQ4 scheme has a larger stencil and the leading truncation error term is bigger. At last, the central variants of these schemes have similar properties to the LSQ schemes built with even polynomial order, meaning that their truncation error only shows terms with even exponent with h and that the central variants order of accuracy is one order higher accurate than their upwind counterparts.

The truncation error analysis of the two-dimensional linear advection equation yields similar results as the ones shown in Table 5 and Table 6 regarding the order of accuracy of the LSQ and S schemes. Nonetheless, LSQ schemes present additional cross-derivatives $x-y$ terms present in its expression of the truncation error. To give an example of this issue the truncation error of the LSQ2 scheme in the two-dimensional linear advection equation is

$$\mathcal{E}_{\text{ULSQ2}}(h) = c_x h^3\left(\frac{19}{84}\left.\frac{\partial^4 u}{\partial x^4}\right\vert_{i,j} + \frac{1}{14}\left.\frac{\partial^4 u}{\partial x^2\partial y^2}\right\vert_{i,j}\right) + c_y h^3\left(\frac{19}{84}\left.\frac{\partial^4 u}{\partial y^4}\right\vert_{i,j} + \frac{1}{14}\left.\frac{\partial^4 u}{\partial x^2\partial y^2}\right\vert_{i,j}\right) + O\left(h^4\right) \quad , \tag{65}$$

where the presence of cross-derivative $x-y$ terms can be clearly observed. This fact is exacerbated as the order of accuracy of the LSQ scheme increases. This allows to conclude that the LSQ schemes in two-dimensional uniform meshes can

Table 6
Leading truncation error term and order of accuracy of different schemes described in Section 3.2.

	$\mathcal{E}$	Order of accuracy
US2	$-\frac{ch^3}{12}\left.\frac{\partial^4 u}{\partial x^4}\right\|_i + O\left(h^4\right)$	3rd
CS2	$-\frac{ch^4}{30}\left.\frac{\partial^5 u}{\partial x^5}\right\|_i + O\left(h^6\right)$	4th
US4	$-\frac{ch^5}{60}\left.\frac{\partial^6 u}{\partial x^6}\right\|_i + O\left(h^6\right)$	5th
CS4	$-\frac{ch^6}{140}\left.\frac{\partial^7 u}{\partial x^7}\right\|_i + O\left(h^8\right)$	6th

present increased numerical errors compared to the one-dimensional LSQ schemes counterparts in the two-dimensional linear advection equation.

It is important to note that the S schemes do not present this cross-derivative terms in the expression of their truncation error. For example, the truncation error of the two-dimensional S2 scheme reads

$$\mathcal{E}_{\mathrm{US2}}\left(h\right) = -\frac{c_x h^3}{12}\left.\frac{\partial^4 u}{\partial x^4}\right|_{i,j} - \frac{c_y h^3}{12}\left.\frac{\partial^4 u}{\partial y^4}\right|_{i,j} + O\left(h^4\right) \quad . \tag{66}$$

It can be observed in the previous expression that S schemes recover the one-dimensional truncation error expression if the solution is uniform along a given principal direction of the structured mesh.

For the sake of brevity, we have decided to not include the leading truncation error terms of the rest of the two-dimensional LSQ schemes, although their order of accuracy is the same as that of their one-dimensional counterparts.

4.2. Two dimensional order of accuracy in non-uniform and unstructured meshes

The order of accuracy of least-squares schemes in unstructured meshes is assessed numerically. The p-th order reconstruction of a given scheme is evaluated by the definition of the order of accuracy or k-exact property. Such an approach is inspired on the work of [36,3]. Previous works studied the k-exact property of the least-squares polynomial and polynomial gradient reconstruction particularized at the cell centers. Nonetheless, the p-th order reconstruction property is extended to analyze the polynomial reconstruction property at quadrature points and face averages reconstructions. A test-polynomial $\mathcal{T}$ is defined as

$$\mathcal{T}\left(\boldsymbol{x}\right) = \left(ax + by\right)^{\mathfrak{p}} \quad , \tag{67}$$

where $a, b \in \mathbb{R}$ and $\mathfrak{p}$ is the polynomial order of the test error function. In particular, the use of different coefficients a and b is justified to avoid the possible mesh principal direction $x = y$. The order of accuracy of a polynomial k-exact scheme can be computed from the greatest value of $\mathfrak{p}$ which ensures that the reconstructed polynomial (built from $\mathcal{T}$) at each cell i follows

$$\mathcal{P}_i\left(\boldsymbol{\xi}\right) = \mathcal{T}\left(\boldsymbol{x}\left(\boldsymbol{\xi}\right)\right) \quad . \tag{68}$$

A k-exact scheme which uses a polynomial of degree p should be capable of reconstructing this test error function with the same or less order $\mathfrak{p} \leq p$ at each point of the cell. In this case the scheme is said to be $p+1$ order accurate. As explained beforehand, the focus is on the k-exact properties of schemes with regards to the interpolation to the quadrature points and the face average reconstructions. To study these properties, two different relative errors will be defined. The first is defined at each face l of the domain and it is computed as

$$\widetilde{\mathcal{E}}_{l,u} = \max\left(\frac{\left\|\widetilde{\mathcal{P}}_\ell^- - \widetilde{\mathcal{T}}_\ell\right\|}{\left\|\widetilde{\mathcal{T}}_\ell\right\|}, \frac{\left\|\widetilde{\mathcal{P}}_\ell^+ - \widetilde{\mathcal{T}}_\ell\right\|}{\left\|\widetilde{\mathcal{T}}_\ell\right\|}\right) \quad , \tag{69}$$

if the studied scheme is upwind and

$$\widetilde{\mathcal{E}}_{l,c} = \frac{\left\|\frac{1}{2}\left(\widetilde{\mathcal{P}}_\ell^- + \widetilde{\mathcal{P}}_\ell^+\right) - \widetilde{\mathcal{T}}_\ell\right\|}{\left\|\widetilde{\mathcal{T}}_\ell\right\|} \quad , \tag{70}$$

for central schemes. In the latter expressions, $\widetilde{\mathcal{P}}_\ell^-$ and $\widetilde{\mathcal{P}}_\ell^+$ refer to the face average of the polynomial build using the central stencil of the face owner and face neighbor cells respectively. These error estimators are related to the numerical error of the computation of the face average conserved variables.

The second relative error used to assess the k-exact properties of the spatial schemes is defined at each quadrature point for upwind schemes as

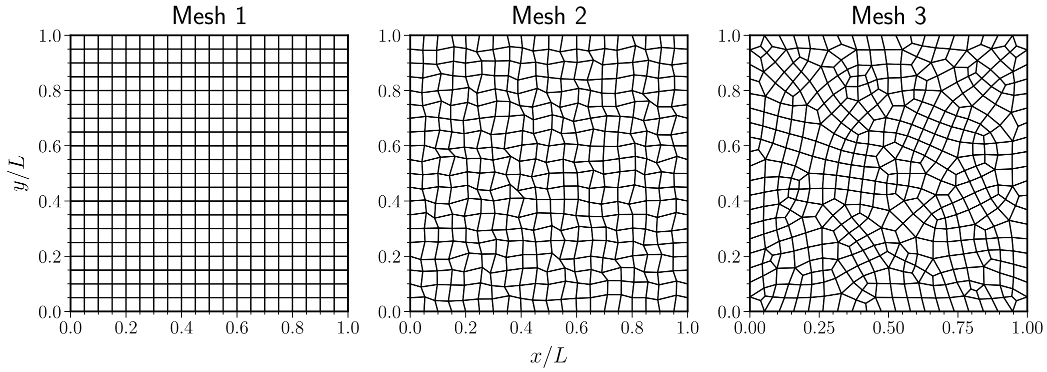


Fig. 4. Grids used to test the k-exact properties of the LSQ schemes.

$$\mathcal{E}_{g,u} = \max\left(\frac{\left\|\mathcal{P}_\ell^-\left(\boldsymbol{\xi}_g\right) - \mathcal{T}\left(\boldsymbol{x}\left(\boldsymbol{\xi}_g\right)\right)\right\|}{\left\|\mathcal{T}\left(\boldsymbol{x}\left(\boldsymbol{\xi}_g\right)\right)\right\|}, \frac{\left\|\mathcal{P}_\ell^+\left(\boldsymbol{\xi}_g\right) - \mathcal{T}\left(\boldsymbol{x}\left(\boldsymbol{\xi}_g\right)\right)\right\|}{\left\|\mathcal{T}\left(\boldsymbol{x}\left(\boldsymbol{\xi}_g\right)\right)\right\|}\right) \quad , \tag{71}$$

and for central schemes as

$$\mathcal{E}_{g,c} = \frac{\left\|\frac{1}{2}\left(\mathcal{P}_\ell^-\left(\boldsymbol{\xi}_g\right) + \mathcal{P}_\ell^+\left(\boldsymbol{\xi}_g\right)\right) - \mathcal{T}\left(\boldsymbol{x}\left(\boldsymbol{\xi}_g\right)\right)\right\|}{\left\|\mathcal{T}\left(\boldsymbol{x}\left(\boldsymbol{\xi}_g\right)\right)\right\|} \quad . \tag{72}$$

This second error estimator is related to the error in the computation of the fluxes at the quadrature points.

Remark 5. It is worth noting that special care should be taken when defining the polynomial test error function to ensure that the minimum absolute value of the polynomial in the mesh is bigger than the machine rounding-off errors by an important margin to avoid divisions by zero in the computation of the numerical error. This is done by displacing the $x-y$ reference frame of the test error function.

Remark 6. Although the test error function is not periodic in the meshes that will be used in this study, the error estimators computation can still be performed with meshes that present periodic boundary faces by ensuring that the periodic ghost cells are initialized with the cell average value of the test error function computed with the geometry of each periodic ghost cell, avoiding the face and quadrature points data exchange due to the mesh periodicity condition. The only drawback of this approach is that the polynomial of the neighbor cells of the faces located in the periodic boundaries cannot be used to compute error estimators. This is due to the fact that the assessment of this polynomial requires periodic data exchange, and as the test error function is not periodic the polynomial reconstruction shows wrong values.

Remark 7. The previous two-dimensional test can be easily expanded to three dimensional analysis by adding the dependence on the z coordinate to the error function. Moreover, it can also be used in one-dimensional cases removing the dependence on the y coordinate.

Three two-dimensional meshes (depicted in Fig. 4) defined in the domain $\boldsymbol{x} \in [0, L]^2$ with periodic boundary condition on all boundaries, are used to compute the previously described error estimators.

1. A uniform structured mesh of edge size h
2. A uniform structured mesh of initial edge size h with a random mesh-nodes perturbation following the expression:

$$\boldsymbol{x}' = \boldsymbol{x} + h\mathcal{K}\,\mathrm{rand}(-1, 1)\ \forall \boldsymbol{x} \in (0, L)^2 \quad , \tag{73}$$

   where $\mathcal{K} = 0.25$ is a constant and the rand function is a random floating point number generator in the domain $[-1, 1] \in \mathbb{R}$. Note that the points which lay on the boundaries of the domain are not modified to respect the periodicity conditions.
3. A fully quad-unstructured mesh with a characteristic edge size of h.

Table 7 reports error estimators in the uniform mesh configuration (Mesh 1) for all the schemes studied in this work and as a function of the test error function order $\mathfrak{p}$. The results show that, as presented in Section 4.1, all schemes are k-exact regarding the interpolation to the quadrature points and the face average reconstruction as they are capable of approximating a test error function with $\mathfrak{p} = p$ up to machine precision rounding-off errors. Moreover, some central scheme variants are k+1-exact in the sense of the face average reconstruction. This property is characteristic of schemes with even polynomial degree p and was also observed in the truncation error analysis (Section 4.1). It is important to note

Table 7
Maximum face average and quadrature points error in the uniform mesh case (Mesh 1) with $h = L/20$. Symbol – denotes errors considered as machine round-off errors ($< 10^{-14}$).

	Maximum Face Average Error			Maximum Quadrature Points Error		
	$\mathfrak{p} = p$	$\mathfrak{p} = p + 1$	$\mathfrak{p} = p + 2$	$\mathfrak{p} = p$	$\mathfrak{p} = p + 1$	$\mathfrak{p} = p + 2$
ULSQ1	–	5.572E–5	1.717E–4	–	5.205E–5	1.607E–4
CLSQ1	–	5.373E–5	1.612E–4	–	5.373E–5	1.655E–4
ULSQ2	–	8.640E–6	3.480E–5	–	9.164E–6	3.696E–5
CLSQ2	–	–	2.276E–7	–	8.183E–6	3.296E–5
ULSQ3	–	1.379E–7	7.198E–7	–	2.017E–7	1.046E–6
CLSQ3	–	1.282E–7	6.411E–7	–	1.282E–7	6.690E–7
ULSQ3W	–	5.042E–8	2.613E–7	–	9.158E–8	4.697E–7
CLSQ3W	–	4.689E–8	2.345E–7	–	4.689E–8	2.429E–7
ULSQ4	–	7.070E–8	4.274E–7	–	8.034E–8	4.866E–7
CLSQ4	–	–	2.884E–9	–	6.458E–8	3.904E–7
US2	–	1.968E–7	7.894E–7	–	2.036E–7	8.166E–7
CS2	–	–	2.241E–9	–	1.925E–7	7.722E–7
US4	–	4.227E–11	2.544E–10	–	4.272E–11	2.571E–10
CS4	–	–	7.606E–13	–	4.075E–11	2.453E–10

Table 8
Maximum face average and quadrature points error in the Mesh 2 configuration with $h \sim L/20$. Symbol – denotes errors considered as machine round-off errors ($< 10^{-14}$).

	Maximum Face Average Error		Maximum Quadrature Points Error	
	$\mathfrak{p} = p$	$\mathfrak{p} = p + 1$	$\mathfrak{p} = p$	$\mathfrak{p} = p + 1$
ULSQ1	–	1.502E–4	–	1.461E–4
CLSQ1	–	8.090E–5	–	1.273E–4
ULSQ2	–	9.648E–6	–	1.019E–5
CLSQ2	–	2.381E–6	–	1.004E–5
ULSQ3	–	3.772E–7	–	4.517E–7
CLSQ3	–	2.051E–7	–	4.047E–7
ULSQ3W	–	1.116E–7	–	1.615E–7
CLSQ3W	–	8.781E–8	–	1.026E–7
ULSQ4	–	8.643E–8	–	9.424E–8
CLSQ4	–	1.650E–8	–	6.978E–8
US2	–	6.574E–8	–	6.653E–8
CS2	–	1.510E–8	–	6.183E–8
US4	–	5.151E–12	–	5.251E–12
CS4	–	1.127E–12	–	5.151E–12

that the previous k+1-exactness is not observed when analyzing the error estimator at the quadrature points. Moreover, every central scheme shows lower error values at the quadrature points than their upwind counterpart. As it will be shown afterwards, this k+1-exactness of some central schemes is lost in non-uniform and unstructured meshes due to the loss of truncation error canceling.

Table 8 presents the values of the different error estimators in Mesh 2 configuration for every scheme studied in this work and as a function of the test error function order $\mathfrak{p}$. As for uniform structured meshes, all upwind LSQ schemes are k-exact schemes regarding the interpolation to the quadrature points and the face average reconstruction, as they are capable to approximate a test error function with $\mathfrak{p} = p$ up to machine precision rounding-off errors in this non-uniform mesh configuration. Moreover, it is possible to observe that the k+1-exactness property of certain central schemes regarding the face average reconstruction is lost in this non-uniform mesh configuration. Nonetheless, it must be noted that the face average error estimator for test error functions such that $\mathfrak{p} = p + 1$ is much lower for the central schemes with even polynomial order p with respect to their upwind counterpart. This difference is less pronounced for least-squares schemes built with odd degree polynomials.

Table 9 shows the different error estimators in the Mesh 3 configuration. The results are very similar compared to those showed in the previous analysis of Table 8 therefore, for the sake of brevity, we avoid its discussion.

Table 9
Maximum face average and quadrature points error in the Mesh 3 configuration with $h \sim L/20$. Symbol — denotes errors considered as machine round-off errors ($< 10^{-14}$).

	Maximum Face Average Error		Maximum Quadrature Points Error	
	$\mathfrak{p} = p$	$\mathfrak{p} = p + 1$	$\mathfrak{p} = p$	$\mathfrak{p} = p + 1$
ULSQ1	–	1.584E–4	–	1.471E–4
CLSQ1	–	7.982E–5	–	1.584E–4
ULSQ2	–	1.139E–5	–	1.180E–5
CLSQ2	–	1.684E–6	–	9.351E–6
ULSQ3	–	4.684E–7	–	6.311E–7
CLSQ3	–	1.936E–7	–	4.684E–7
ULSQ3W	–	9.529E–8	–	1.536E–7
CLSQ3W	–	7.387E–8	–	7.984E–8
ULSQ4	–	6.802E–8	–	8.250E–8
CLSQ4	–	1.662E–8	–	6.450E–8

5. Spectral properties of finite volume schemes

In the following, spectral properties of the numerical reconstruction of the conservation equation (10) will be addressed. Let us assume a linear flux operator $\mathcal{F}(u) = \boldsymbol{c}u$, where $\boldsymbol{c}$ is the convective speed. In particular, Eq. (10) discretized in a mesh of $N_{\mathcal{V}}$ cells, may be recast to a dynamical system [10] of

$$\frac{\mathrm{d}\underline{\overline{u}}^{\delta}}{\mathrm{d}t} = \boldsymbol{J}\underline{\overline{u}}^{\delta} \quad , \tag{74}$$

where $\boldsymbol{J} \in \mathbb{R}^{N_{\mathcal{V}} \times N_{\mathcal{V}}}$ is a matrix which depends on the upwinding parameter f of each numerical scheme and it is constructed using the interpolation polynomials. Therefore, it uniquely depends on geometric properties of the mesh discretization. In addition, $\underline{\overline{u}}^{\delta} \in \mathbb{C}^{N_{\mathcal{V}}} = \left[\overline{u}_1^{\delta}, \overline{u}_2^{\delta}, \ldots, \overline{u}_i^{\delta}, \ldots \overline{u}_{N_{\mathcal{V}}}^{\delta}\right]$ is a vector containing cell average values of the solution at each cell. Von Neumann analysis takes advantage of the particular structural properties of the matrix $\boldsymbol{J}$, i.e. circulant, orthogonal, etc. For instance in the Toeplitz case, its symbol provides access to the spectrum and the pseudo-spectrum (see [41]). Nonetheless, the general case is slightly more complex. Toeplitz property is lost whenever the underlying numerical triangulation is not uniform, in such a case entries of the matrix $\boldsymbol{J}$ cease to be in a particular ordering.

Integration of the dynamical system Eq. (74) can be analytically performed with a semi-group propagator, i.e. the exponential matrix, so one ends up with a discrete map as follows

$$\underline{\overline{u}}^{\delta}(t_{n+1}) = \boldsymbol{R}(\tau, \boldsymbol{J})\,\underline{\overline{u}}^{\delta}(t_n) \quad , \tag{75}$$

where $\boldsymbol{R}(\tau, \boldsymbol{J}) = \boldsymbol{R}^{\delta} = e^{\tau \boldsymbol{J}}$ is the exponential of the matrix $\boldsymbol{J}$, t_n is the solution at a given discrete time and $t_{n+1} - t_n$ is the time step τ. Similarly, one could choose an approximation of the actual semi-group, i.e. a Runge-Kutta multistage integrator $\mathbf{R}^{\delta}$ with a given number of stages and order. Numerical examples are carried out with a RK33 temporal discretization method. This method allows to compute $\mathbf{R}^{\delta}$ as

$$\boldsymbol{R}^{\delta} = I + \tau \boldsymbol{J} + \frac{\tau^2}{2}\boldsymbol{J}^2 + \frac{\tau^3}{6}\boldsymbol{J}^3 \quad . \tag{76}$$

Furthermore, let us remind that the semi-group of the linear advection equation with periodic boundary conditions and initial condition:

$$u(\boldsymbol{x}, t = 0) = \hat{u}\, \mathrm{e}^{\mathrm{I}\kappa\left(x\cos\theta + y\sin\theta\right)} \quad , \tag{77}$$

where κ is the spatial wavenumber, is analytical with the following expression: $\boldsymbol{R} = e^{-\mathrm{I}\omega\tau} I$, such that $\omega = \kappa\left(c_x \cos\theta + c_y \sin\theta\right)$ is the analytical temporal wavenumber. In this work we will only consider advection velocity vectors such that $\zeta = \pi/4$.

Remark 8. If the exponential operator is used to build the matrix operator $\boldsymbol{R}$, the results will only take into account spatial discretizations defects. On the contrary, the consideration of Runge-Kutta multistage integrators to build $\boldsymbol{R}^{\delta}$ imply that the numerical solution will include both spatial and temporal defects.

5.1. Von Neumann analysis in structured uniform meshes

Von Neumann analysis on structured uniform computational domains with periodic boundary conditions takes advantage of the circulant property of the linear operator $\boldsymbol{J}$, i.e. the action matrix-vector product is equivalent to the convolution of two vectors, thus the Fourier transform is a pointwise multiplication. These properties can only be obtained if the mesh is uniform, the stencils of each cell have the same topological shape and the boundary conditions are periodic. In addition, it is possible to show the equivalency between an analysis of Eq. (75) in the finite-volume formulation and in the finite-difference sense. Thus, Eq. (75) can be recast into:

$$\underline{u}(t_{n+1}) = \boldsymbol{R}^{\delta}\,\underline{u}(t_n) \quad , \tag{78}$$

where $\underline{u} \in \mathbb{C}^{N_{\mathcal{V}}}$ represents the point-value solution at the cell centroids. As matrix $\boldsymbol{J}$ is circulant, so is $\boldsymbol{R}^{\delta}$. Von Neumann analysis aims to determine the deviation of the numerical solution to Eq. (78) and the solution to the advection equation. Let us consider the solution to the advection equation

$$u(\boldsymbol{x}, t) = \hat{u}\, \mathrm{e}^{\mathrm{I}\kappa(x\cos\theta + y\sin\theta) - \mathrm{I}\omega t} \quad , \tag{79}$$

which is injected to Eq. (78), to obtain the value of the numerical temporal wavenumber of each cell

$$\omega_i^{\delta} = \omega^{\delta} = \frac{\mathrm{I}}{\tau}\ln\left(\sum_j \boldsymbol{R}^{\delta}_{i,j}\mathrm{e}^{\mathrm{I}\kappa\left((x_j - x_i)\cos\theta + (y_j - y_i)\sin\theta\right)}\right) = \mathrm{I}\frac{\ln Q}{\tau} \quad , \tag{80}$$

where ω^{δ} is the numerical temporal wavenumber and Q is an amplification factor. It is emphasized that this numerical temporal wavenumber has the same value for each cell due to the properties of the $\boldsymbol{R}^{\delta}$ matrix. This uniqueness will be lost in unstructured or non-uniform meshes as it will be shown afterwards.

To assess the dissipation and dispersion errors, a non-dimensional temporal wavenumber error function is defined as

$$\mathcal{D}(\kappa h, \mathrm{CFL}, \theta) = \frac{1}{\omega}\left(\left|\mathrm{Re}\left(\omega^{\delta} - \omega\right)\right| + \mathrm{I}\left|\mathrm{Im}\left(\omega^{\delta} - \omega\right)\right|\right) \quad , \tag{81}$$

where $\mathrm{CFL} = \frac{c\tau}{h}$. The comparison of the numerical temporal wavenumber and its analytical value is commonly used to assess the dissipation ($\mathrm{Im}(\mathcal{D})$) and dispersion ($\mathrm{Re}(\mathcal{D})$) errors of the numerical schemes [42–46]. The one-dimensional case can be particularized by imposing $\theta = \zeta = 0$ and by using a one-dimensional mesh.

If temporal discretization errors are neglected (by assuming exponential temporal integration), it is possible to obtain an analytical expression of the non-dimensional error function. For example for the one-dimensional ULSQ1 scheme this error function can be written as

$$\mathcal{D}(\kappa h) = -\frac{(\kappa h)^2}{12} + \mathrm{I}\frac{(\kappa h)^3}{8} + O\left(\kappa h\right)^4 \quad . \tag{82}$$

In the latter formula, it is possible to differentiate the dissipation (imaginary part) and dispersion errors (real part) of the given spatial discretization scheme. Moreover, the leading error term has an exponent which is of the same order of the truncation error term in Table 5. Therefore, the truncation error terms with odd exponent produce dissipation and the ones with even exponent produce dispersion. It is also important to note that central schemes do not present dissipation errors when using analytical integration methods due to the fact that their matrix $\boldsymbol{J}$ is anti-symmetric. This implies that the eigenvalues of their matrix $\boldsymbol{J}$ are purely imaginary. Hence, dissipation of central schemes can uniquely appear due to the temporal numerical integration method. This can also be observed by analyzing the truncation errors of central schemes showed in Table 5.

5.1.1. One-dimensional von Neumann analysis

The one-dimensional von Neumann analysis can be performed by supposing a plane wave and a uniform periodic mesh along the x direction with constant cell size h. This analysis will be performed using the cells per wavelength parameter $\lambda/h = \frac{2\pi}{\kappa}$ (similar to the points per wavelength parameter used in [47]) instead of the more broadly used non-dimensional spatial wavenumber parameter κh.

Dissipation & dispersion

Dissipation and dispersion of finite volume schemes are analyzed by considering the real and imaginary parts of $\mathcal{D}$. The study will be performed with three different CFL values. The lowest $\mathrm{CFL} = 0.1$ is chosen to ensure that the spatial discretization errors are dominant; the highest $\mathrm{CFL} = 1.0$ is chosen to evaluate the numerical errors of simulations dominated by temporal discretization errors. At last, the case with $\mathrm{CFL} = 0.5$ will also be presented to assess the interactions between the spatial and temporal discretization errors.

Fig. 5 and Fig. 6 display the dispersion error of the upwind and central schemes respectively. The figures show that for all CFL values the upwind and central scheme variants present a similar trend. Let us start by considering dispersive errors

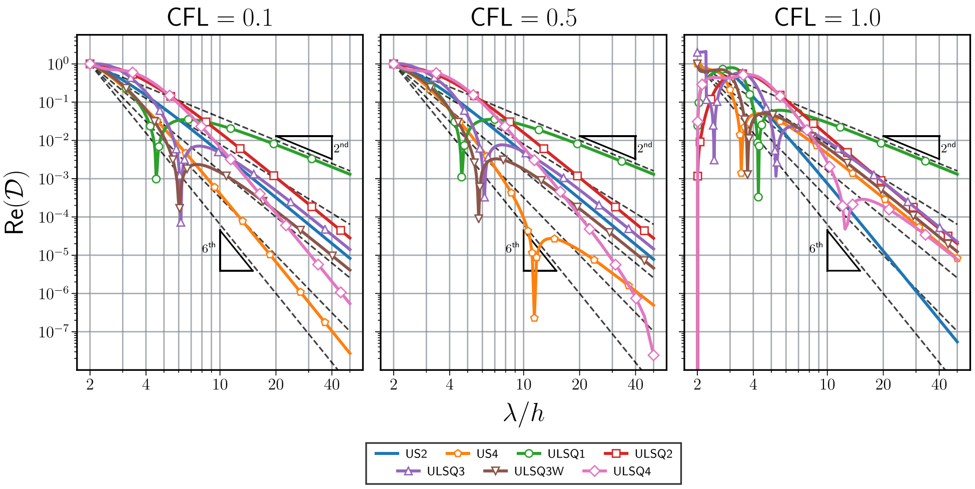


Fig. 5. Dispersion errors of 1D upwind schemes as a function of λ/h and for CFL $\in [0.1, 0.5, 1.0]$. Dashed lines are related to the function $\mathrm{Im}(D) \sim (\lambda/h)^n$ where n is an exponent ranging from 2 to 6 and are used to assess the leading term order of the dispersion errors.

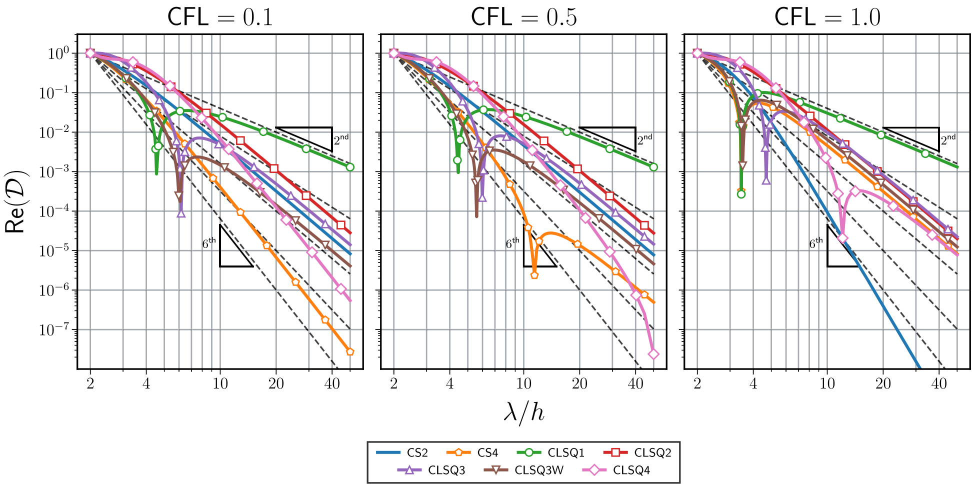


Fig. 6. Dispersion errors of 1D central schemes as a function of λ/h and for CFL $\in [0.1, 0.5, 1.0]$. Dashed lines are related to the function $\Im(D) \sim (\lambda/h)^n$ where n is an exponent ranging from 2 to 6 and are used to assess the leading term order of the dispersion errors.

at CFL = 0.1. The dispersion error evolves as a power law with the number of cells per wavelength which is representative of even exponents of the truncation error terms analyzed in Section 5.1. Nonetheless, it is worth noting that such a trend appears at values of $\lambda/h > 10$, whereas at lower values most least-squares schemes do not exhibit this behavior. Let us stress out that even if the dispersion errors follow the correctly predicted trend for high values of λ/h, error values of most LSQ schemes are higher than those of the corresponding structured (S) schemes. An exception to this behavior is observed when analyzing the LSQ3W scheme which shows lower dispersion errors than the S2 scheme with the same order of accuracy.

For high CFL values most upwind and central schemes show similar dispersion errors and a 4th order trend with h. This can be explained due to the temporal discretization errors, which in our case, due to constant CFL condition, behave as $O(\tau^3) \sim O(h^3)$. As dispersion errors are caused by terms with even exponent truncation errors with h, the dispersion error will behave as the 4th order truncation error of the temporal discretization method $O(\tau^4) \sim O(h^4)$.

It is also interesting to comment the special behavior of the LSQ1 and S2 schemes as the CFL increases. Scheme LSQ1, does not display a 4th order trend with h because this scheme has a 2nd order in space which is lower than the order in time, therefore its dispersive errors are not influenced by temporal discretization. On the other hand, the S2 scheme shows a decrease of its dispersion errors as the CFL increases. This fact is exacerbated in its central variant scheme CS2, in which we observe a very high-order trend of the dispersion errors with the cell size parameter. The reason for such superconvergence behavior for CFL = 1 was not identified. This superconvergence has not been observed when combining the S2 scheme with other RK time integrators at CFL = 1. Therefore, further studies are needed to better understand this observation.

Fig. 7 displays the evolution of the dissipation error with respect to λ/h for the upwind schemes. Dissipation error of each upwind scheme shows a similar trend to dispersion error. For low CFL values, dissipation is dominated by spatial

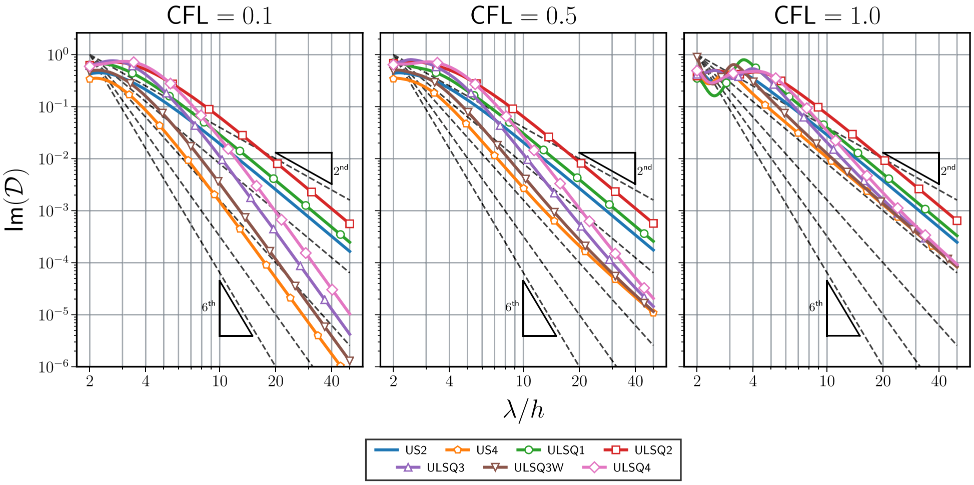


Fig. 7. Dissipation errors of 1D upwind schemes as a function of λ/h and for CFL $\in [0.1, 0.5, 1.0]$. Dashed lines are related to the function $\mathrm{Im}(D) \sim (\lambda/h)^n$ where n is an exponent ranging from 2 to 6 and are used to assess the leading term order of the dissipation errors.

discretization errors, whereas at large CFL the opposite holds. At low CFL, a power-law trend of the dissipation error is observed for high values of the number of cells per wavelength. The exponent of the power law corresponds to the odd exponent of the truncation error of each scheme. For low values of λ/h the least-squares schemes underperform compared to the US2 and US4 schemes. In particular, we remark that the ULSQ3W over-performs every other least-squares scheme and the S2 scheme for each λ/h, which shows that the addition of distance weighting to the least-squares schemes may improve the dissipation properties of these schemes. As the CFL number increases temporal discretization errors dominate for high resolved waves. Under those circumstances, every scheme displays a $O\left(h^3\right)$ error trend, similar to what it has been observed with the dissipation of central schemes.

The dissipation errors of central schemes are not shown due to the fact they are just given by the 3rd numerical error of the RK33 time integrator, due to the choice of constant CFL. Such behavior is equally observed for high values of the CFL number and high values of the cells per wavelength for upwind schemes, i.e. in these cases the temporal discretization error dominates.

Temporal stability

This section is devoted to the study of the L^2-stability of the linear advection equation, thus maximum CFL values will be reported for every spatial numerical scheme presented in this study. Maximum CFL is assessed from Eq. (80), by taking into account an amplification factor lower than one

$$\| Q \| \leq 1 \quad , \tag{83}$$

which is equivalent to stating that the absolute part of every eigenvalue of operator $\boldsymbol{R}^{\delta}$ is lower than one.

The eigenvalues of $\boldsymbol{R}^{\delta}$ vary with the CFL and λ/h. Therefore, for each value of κh there is a given CFL which ensures linear stability. The lowest value of these CFL conditions, denoted as $\mathrm{CFL_{MAX}}$ and is the maximum CFL value and ensuring linear stability for a given scheme.

Table 10b reports the maximum CFL number ensuring linear stability for the S2 and S4 schemes in a uniform mesh using the RK33 time integration method. Let us remark the validity of the approach which has been validated via a cross-comparison of the maximum CFL of US4 scheme, previously reported by [1]. It can be observed that S2 schemes present higher $\mathrm{CFL_{MAX}}$ than S4 schemes. Additionally, structured central schemes display a lower $\mathrm{CFL_{MAX}}$ than their upwind counterparts. Such a behavior is expected as the central schemes have much lower dissipation values than the upwind schemes and therefore, the linear stability can only be ensured for lower values of the CFL number.

Table 10a reports the maximum CFL number ensuring linear stability for the different one dimensional least-squares schemes in a uniform mesh using the RK33 time integration method. The linear stability CFL criterion of these schemes behaves differently to that of the S2 and S4 schemes and does not follow a coherent trend with respect to their order of accuracy and stencil size. Moreover, one may also observe that for most LSQ schemes, the central variant displays a higher value of the $\mathrm{CFL_{MAX}}$ than the upwind variant.

5.1.2. Two-dimensional analysis

The two-dimensional von Neumann analysis is carried out with the same tools developed for the one-dimensional case. However, the dissipation and dispersion maps depend on the wave angle θ and the advection velocity vector. For these studies, the convection velocity angle ζ is set as $\zeta = 45°$. This convection velocity angle ensures that the dissipation and dispersion spectrums are computed in the preferential direction of propagation for quadrilateral structured uniform meshes

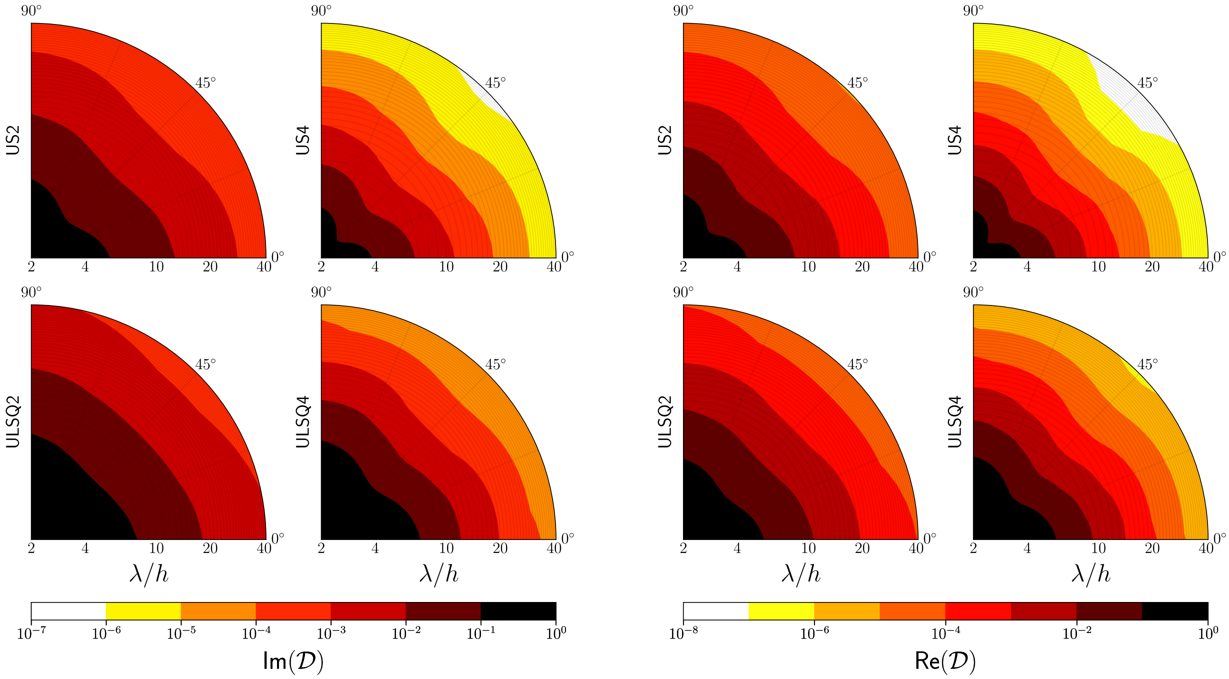
Table 10
Maximum CFL value to ensure temporal stability in the one-dimensional linear-advection as a function of the different schemes studied in this work. Results obtained with the RK33 time integration scheme.

Scheme	$\mathrm{CFL}_{\mathrm{MAX}}$
ULSQ1	1.176
CLSQ1	1.101
ULSQ2	1.353
CLSQ2	1.666
ULSQ3	1.312
CLSQ3	1.342
ULSQ3W	1.434
CLSQ3W	1.111
ULSQ4	1.292
CLSQ4	1.718

(a) One-dimensional LSQ schemes

Scheme	$\mathrm{CFL}_{\mathrm{MAX}}$
US2	1.626
CS2	1.262
US4	1.435
CS4	1.092

(b) One-dimensional S schemes



(a)

(b)

Fig. 8. Diffusion (a) and Dispersion (b) errors of the two-dimensional US2, US4, ULSQ2 and ULSQ4 schemes at $\zeta = 45°$. The radial coordinate of the figures represents the cells per wavelength λ/h parameter, while the azimuthal coordinate is the wave angle. Results are obtained for CFL $= 0.05$. (For interpretation of the colors in the figure(s), the reader is referred to the web version of this article.)

(see [48,49]). Moreover, as dissipation and dispersion depend on the wave angle θ, these parameters need to be analyzed for each wave angle. The temporal discretization time step is fixed such that $c\tau/h = 0.05$, as we have observed that the influence of the temporal discretization methods in the von Neumann analysis is similar to the one-dimensional analysis. We will consider that the computation of the $\mathrm{CFL}_{\mathrm{MAX}}$ condition of the schemes in two-dimensional problems is beyond the scope of this work, although the methods developed in the previous sections are still valid to compute $\mathrm{CFL}_{\mathrm{MAX}}$. The interested reader is referred to [46] for further comments regarding the extension of the $\mathrm{CFL}_{\mathrm{MAX}}$ condition to two-dimensional problems.

Fig. 8a show the contours of the dissipation for the upwind US2, US4, ULSQ2 and ULSQ4 schemes as a function of the number of cells per wavelength and the wave angle. These contours show clearly that dissipation is always minimum at $\theta = 45° = \zeta$ for all schemes. This behavior was also described in [42]. Moreover, the dissipation values show similar trends as in the one-dimensional studies. The ULSQ2 is more dissipative than the US2 scheme, although they have the same order of accuracy. The previous statement can also be applied when comparing the ULSQ4 and US4 schemes. At last, we observe that the ULSQ4 scheme shows very high values of the dissipation error compared to the other schemes for under-resolved

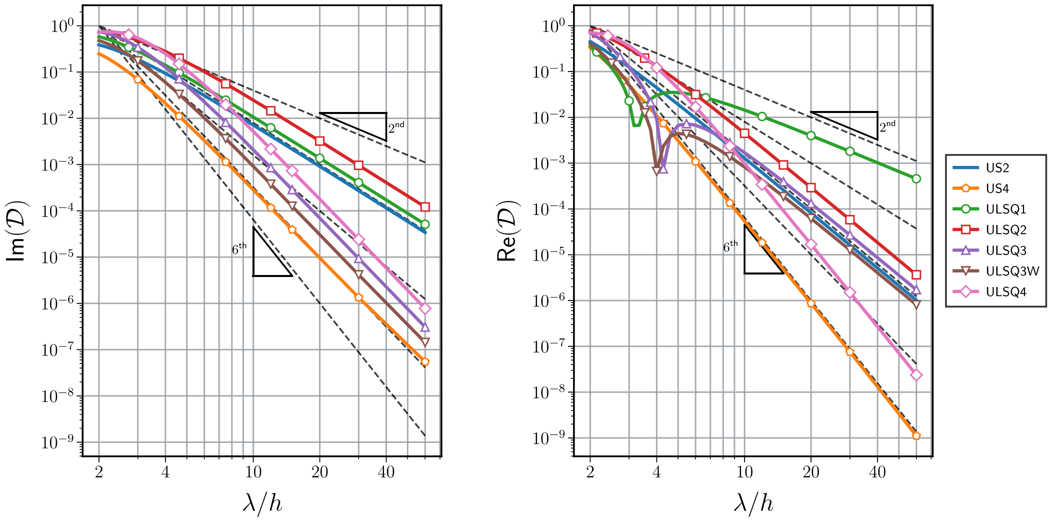


Fig. 9. Dissipation and dispersion errors of 2D upwind schemes of waves with $\theta = 45°$ as a function of λ/h and for $c\tau/h = 0.05$. Dashed lines are related to the function $\mathrm{Im}(D) \sim (\lambda/h)^n$ where n is an exponent ranging from 2 to 6 and are used to assess the leading term order of the dissipation errors.

waves in this two-dimensional analysis. This behavior was also noted in the one-dimensional dissipation analysis. Similar conclusions can be obtained regarding the dispersion errors as shown in Fig. 8b.

For comparison purposes, Fig. 9 shows the dissipation and dispersion errors of all two-dimensional upwind schemes at $\theta = 45°$. The results obtained are very close to those obtained in the one-dimensional study at low CFL number (Fig. 7 and Fig. 5), therefore we avoid its discussion. It is also observed that the two-dimensional central schemes behave similarly to their one-dimensional counterparts. This implies that the dissipation of these schemes is dominated by the temporal discretization errors and that their dispersion is equivalent to the dispersion of the upwind schemes for low CFL numbers.

We avoid adding more two-dimensional comparisons for the sake of brevity. This is justified due to the fact that the one-dimensional conclusions regarding the numerical dissipation and dispersion can also be applied to the two-dimensional analysis for all the schemes.

5.2. Diffusion & dispersion errors in two-dimensional unstructured grids

The von Neumann analysis of Section 5.1 can be further expanded to study dissipation and dispersion errors in unstructured meshes. It is important to note that in non-uniform meshes, the $\boldsymbol{J}$ matrix is not Toeplitz nor circulant, therefore the properties that allowed to simplify the numerical error analysis in the von Neumann method cannot be used. This is due to the fact that circulant matrices' eigenvectors are Fourier modes and therefore, the dissipation and dispersion properties of a wave described by a Fourier mode are just given by a single eigenvalue of $\boldsymbol{J}$. To extend the von Neumann analysis for unstructured meshes, we first define the cell average of a given two-dimensional wave function

$$\overline{u}_i(t) = E_i \hat{u}_i(t) \quad , \tag{84}$$

where $\hat{u}_i(t=0) = \text{cte}$ and $E \in \mathbb{C}^{N_\mathcal{V} \times N_\mathcal{V}}$ is a diagonal matrix whose entries E_i are defined as

$$E_i = \frac{1}{|\mathcal{V}_i|} \int\limits_{\mathcal{V}_i} e^{\mathrm{I}\kappa(x\cos\theta + y\sin\theta)} \, \mathrm{d}\mathcal{V} \quad . \tag{85}$$

The linear system in Eq. (75) is then expressed as

$$\underline{\hat{u}}(t_{n+1}) = E^{-1} \boldsymbol{R}^\delta E \; \underline{\hat{u}}(t_n) \quad , \tag{86}$$

where $\underline{\hat{u}} \in \mathbb{C}^{N_\mathcal{V} \times N_\mathcal{V}}$ is a vector containing the values of $\hat{u}_i$ at each cell. If the initial vector $\underline{\hat{u}}(t_n)$ is an eigenvector of $E^{-1}\boldsymbol{R}^\delta E$, then the solution of the previous equation is given by $\underline{\hat{u}}(t_{n+1}) = \Lambda \, \underline{\hat{u}}(t_n)$, where Λ is an eigenvalue. This is typically the case for the von Neumann analysis in uniform meshes. However, in non-uniform meshes the previous statements do not hold true and therefore, the vector $\underline{\hat{u}}(t_{n+1})$ will not only depend on a single eigenvalue, but on a combination of them. This also implies that $\underline{\hat{u}}(t_{n+1})$ won't be uniform even if the initial $\underline{\hat{u}}(t_n)$ is uniform. Therefore discrete system response described in Eq. (86) will present a transient regime. As a result, the dissipation and dispersion analysis in non-uniform meshes has to be performed using global estimators under asymptotic behavior assumptions. To do so, one can build a numerical estimator of the imaginary (dissipation) and real (dispersion) parts of the temporal wavenumber as

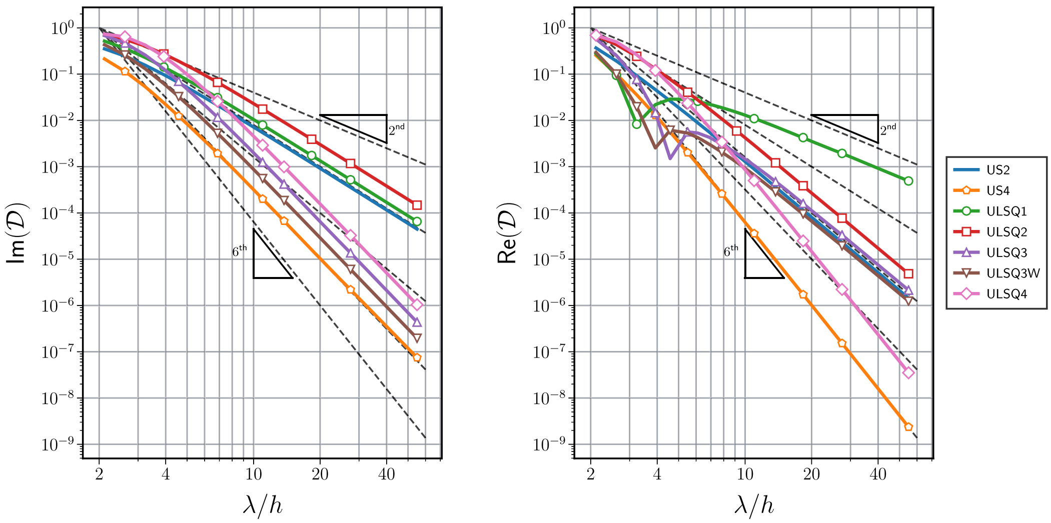


Fig. 10. Dissipation and dispersion errors of 2D upwind schemes of waves with $\theta = 45°$ in Mesh 2 configuration as a function of λ/h and for $c\tau/h = 0.05$. Dashed lines are related to the function $\mathrm{Im}(D) \sim (\lambda/h)^n$ where n is an exponent ranging from 2 to 6 and are used to assess the leading term order of the dissipation and dispersion errors.

$$\mathrm{Im}(\omega_n^\delta) \approx \frac{1}{\tau} \ln \frac{\|\hat{\underline{u}}(t_{n+1})\|}{\|\hat{\underline{u}}(t_n)\|} \quad , \tag{87}$$

and

$$\mathrm{Re}(\omega_n^\delta) \approx -\frac{1}{\tau} \arg \langle \hat{\underline{u}}(t_{n+1}), \hat{\underline{u}}(t_n) \rangle \quad . \tag{88}$$

A posteriori analyses of the dissipation and dispersion estimators have confirmed that the values of the dissipation and dispersion estimators at $n = 0$ yield accurate predictions of the numerical errors observed in the simulation of the linear advection in two-dimensional non-uniform and/or unstructured meshes (see Section 6.3). This fact was also recently discussed in [50], where it was stated that the *short-term* numerical errors of Spectral Element Methods could be predicted by analyzing the dissipation estimator (Eq. (87)) at $n = 0$. However, it is worth noting that the dissipation and dispersion properties of badly-resolved wavelengths when increasing the iteration number n. This issue appears due to the initial high dissipation values of under-resolved waves and machine rounding-off errors. The previous observations give enough evidence that the analysis of the global estimator of the numerical temporal wave number can be made for the first iteration $n = 0$ at least for numerical schemes and meshes studied in this work. As explained earlier, these meshes present close-to-uniform cell sizes with moderate cell aspect ratio changes. It has been observed, a posteriori, that meshes with substantial cell aspect ratio variation exhibit a considerably different eigenvalue spectrum from that obtained with the meshes used in this work. Such eigenvalues distribution has an important impact the linear dissipation and dispersion transient behavior, possibly invalidating the conclusions of this work.

At last, we stress out that these observations only apply to the mesh of this study. The conclusions obtained from this section, may not be the extrapolated to meshes with abruptly varying aspect ratios or bigger skewness factor or with non-linear spatial discretization schemes.

Under the previous assumptions, equations (87) and (88) (particularized for $n = 0$) can be introduced in the temporal wavenumber error function $\mathcal{D}$ (Eq. (81)) to compute the dissipation and dispersion errors of the global estimator of numerical temporal wavenumber.

It should be noted that the previous method is analogous to the von Neumann analysis in uniform meshes, as in those meshes there are no transient effects due to the properties of circulant matrices.

The previously discussed method can be applied to study dissipation and dispersion properties of unstructured meshes. It is important to note that this method cannot be applied to study arbitrary waves, as the waves need to respect the following periodicity constraints of the domain

$$\kappa \cos\theta = \frac{2\pi n_x}{L} \text{ and } \kappa \sin\theta = \frac{2\pi n_y}{L} \quad , \tag{89}$$

where n_x and n_y are positive integer numbers. In uniform meshes, this particularization was not needed as one can assume that the meshes have arbitrary domain and cell sizes which respect the periodicity condition.

Fig. 10 and Fig. 11 show the dissipation and dispersion errors of the upwind least-squares schemes for the Mesh 2 and Mesh 3 configurations at $n = 0$ and $c\tau/h = 0.05$. Both figures show similar results. The dissipation and dispersion properties of the least-squares schemes in the non-uniform meshes of this study are very similar to those found in uniform meshes.

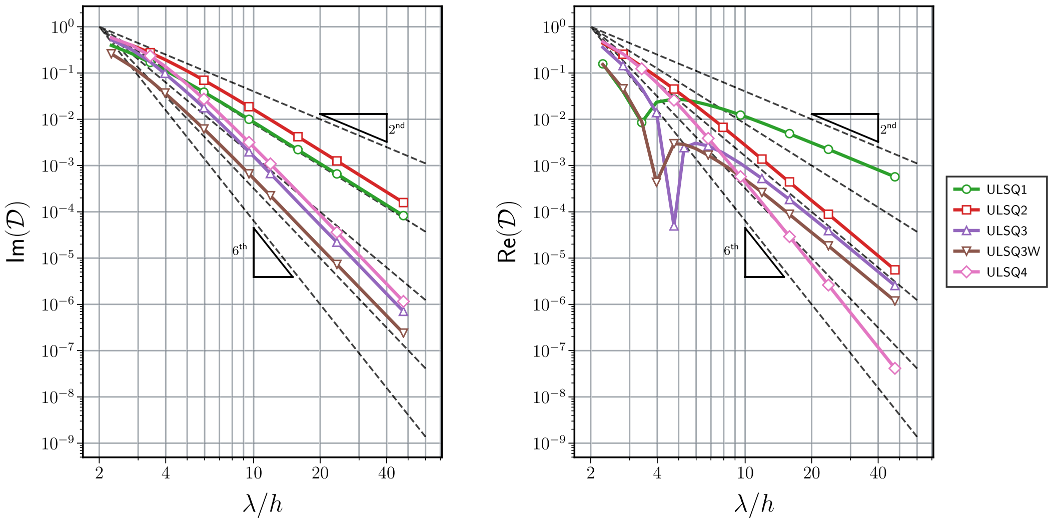


Fig. 11. Dissipation and dispersion errors of 2D upwind schemes of waves with $\theta = 45°$ in Mesh 3 configuration as a function of λ/h and for $c\tau/h = 0.05$. Dashed lines are related to the function $\mathrm{Im}(D) \sim (\lambda/h)^n$ where n is an exponent ranging from 2 to 6 and are used to assess the leading term order of the dissipation and dispersion errors.

This leads to the conclusion that high-order least-squares schemes are robust regarding their accuracy and numerical properties when meshes lose their global uniformity. This is not generally the case for other linear reconstruction schemes, which might lose accuracy and spectral properties with mesh deformation or perturbation. Moreover, this reveals that most conclusions obtained in the one-dimensional analysis remain valid. This allows to conclude that the one-dimensional analysis of least-squares schemes is representative of their behavior in the non-uniform two-dimensional meshes described in this work. Such meshes present a certain degree of cell size uniformity which, as described previously, does not substantially modify the eigenvalues spectrum with respect to that of uniform meshes. We omit central schemes from the comparison due to the fact that their $\boldsymbol{J}$ operator has eigenvalues with positive real part in the aforementioned configurations and therefore those schemes are inherently unstable.

6. Numerical experiments

In this section, numerical simulations will be performed to validate the previous analytical analysis in linear and non-linear cases.

6.1. Error computation

To assess the numerical errors of the simulations the L_∞ norm based on the cell averages of the numerical will be used. In our study, this norm is defined as

$$L_\infty(u,t) = \max_{i=1}^{N_\mathcal{V}} \left(|\overline{u}_i^\delta(t) - \overline{u}_i(t)| \right) \quad . \tag{90}$$

The computation of this error norm with different number of cells in the domain will allow to perform an a posteriori error norm convergence order study. The order of the error norm convergence should be consistent with the order of the truncation errors studied in Section 4.1.

6.2. Linear advection in one-dimensional mesh

In this section, the one-dimensional linear advection equation (Section 2.5.1) is solved using the LSQ and S based finite-volume schemes described in Section 3.1 and Section 3.2. In this analysis, the initial condition is defined by a smooth sine wave [45]:

$$u(x, t = 0) = \sin\left(2\pi \frac{x}{\lambda}\right), \quad x \in [0, L] \quad . \tag{91}$$

In the previous initial condition, we fix $\frac{L}{\lambda} = 4$, $L = 1$ and $c = 1$. Moreover, we define the non-dimensional parameter $\frac{cT}{L} = 1$ where T is the wave temporal period in the domain. The domain is uniformly discretized with cell size h and periodic boundary conditions at $x = 0$ and $x = L$ are applied. To assess the numerical properties of each numerical scheme, we run

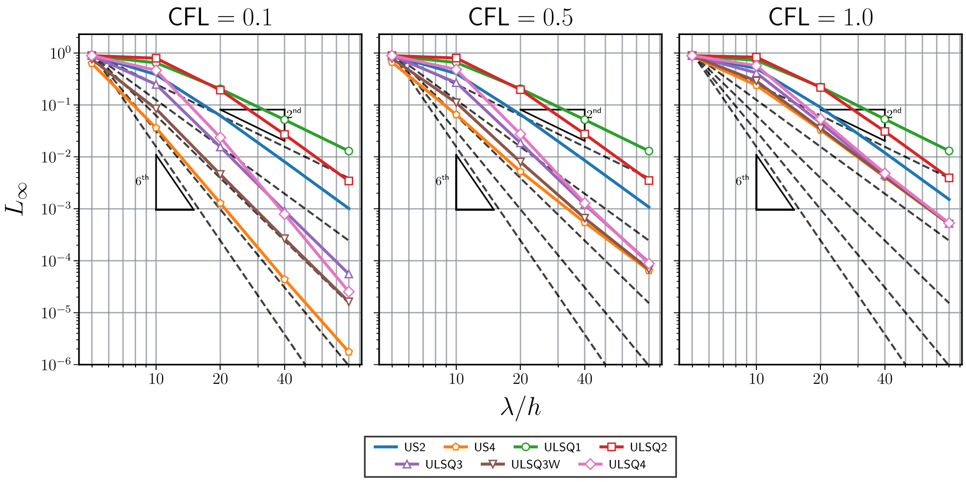


Fig. 12. L_∞ error norm of the solution in the 1D test case at $t = T$ using upwind schemes for three different values of the CFL and as a function of λ/h.

simulations at fixed $\mathrm{CFL} = c\tau/h = 0.1, 0.5$ and 1.0 while decreasing the cell size h (which implies implicitly a reduction of the time step τ). The numerical errors will be analyzed at $t = T$ using the L_∞ norm (Eq. (90)).

Fig. 12 shows the temporal evolution of the L_∞ norm of the upwind schemes as a function of the non-dimensional time t/T, different values of λ/h and the three CFL values mentioned. These results are representative of the combination of the dissipation and dispersion of the spatial and temporal discretization schemes. In this figure, it is possible to observe that error norm of all the spatial discretization schemes shows their predicted theoretical order of accuracy trend (as discussed in Table 5 and Table 6 for $\mathrm{CFL} = 0.1$). However, as the CFL value increases, the higher order schemes show much higher errors due to the temporal discretization schemes, degrading their order of accuracy to a maximum of three as predicted in Section 5.1. To assess the difference spatial discretization schemes, we focus on the results with the lower CFL value. Among LSQ schemes, the ULSQ3W scheme outperforms all others in the studied range of λ/h, although the ULSQ4 (which has a bigger stencil) should yield better results than the ULSQ3W for $\lambda/h > 80$ due to its order of accuracy which is one order higher. The error values of the ULSQ3W are three times lower than that of its counterpart ULSQ3 schemes for well-resolved waves. This is consistent with the reduction of the magnitude of the truncation error shown in Table 5. It is worth noting that the ULSQ4 scheme behavior with badly resolved waves is closer to that of the lower order ULSQ1 and ULSQ2 schemes. This might indicate that ULSQ schemes with large stencils might need additional treatment (such as the addition of distance weighting methods described in Section 3.1.3) to better handle low resolution waves.

Regarding the S schemes, they present much lower error norm values than LSQ schemes with the same stencil or with the same order of accuracy. In particular, the US2 scheme, which is comparable to the ULSQ1 and ULSQ2 schemes in terms of stencil size and order of accuracy respectively, shows one more order of accuracy and five times less error than the ULSQ1 and ULSQ2 schemes respectively. Moreover, the US4 scheme, which can be compared to the ULSQ2 and ULSQ4 schemes in terms or order of accuracy, clearly outperforms both by large orders of magnitude. It is also worth noting that the S schemes outmatch the LSQ schemes in the prediction of the propagation of low resolution waves.

Similar conclusions can also be made regarding the errors obtained when using central schemes as shown on Fig. 13. However, due to the truncation error cancellation properties of the central schemes built with even polynomial order, it is possible to note that the CLSQ2, CLSQ4, CS2 and CS4 schemes clearly outperform their upwind variants. Nonetheless, the CLSQ1, CLSQ3 and CLSQ3W show similar results to their upwind variants.

6.3. Linear advection in two-dimensional mesh

To test the numerical properties of the finite volume schemes in 2D linear cases the two-dimensional linear advection equation is solved with the following initial wave condition used in [51]

$$u(x, y, t = 0) = \sin\left(2\pi \frac{x}{\lambda}\right) \cos\left(2\pi \frac{y}{\lambda}\right), \quad \boldsymbol{x} \in [0, L]^2 \quad , \tag{92}$$

where $\lambda = L/4$ is a given characteristic wavelength, $L = 1$ and $c_x = c_y = 1$. We define the non-dimensional parameter $\frac{c_x T}{L} = 1$ where T is the wave temporal period. To study this case, the three difference meshes discussed in Section 4.2 with characteristic cell size h will be used. Periodic boundary conditions are applied at each edge of the domain. To assess the numerical properties of each numerical scheme we run simulations at fixed $c_x\tau/h = 0.05$ while decreasing the cell size parameter h, so as to ensure that spatial discretization errors are dominant with respect to temporal discretization errors.

Fig. 15a and Fig. 14b show the error norm for Mesh 1 configuration (uniform mesh) and the upwind and central schemes respectively. The results are very close to those shown in the one-dimensional analysis of Section 6.2, as was also predicted

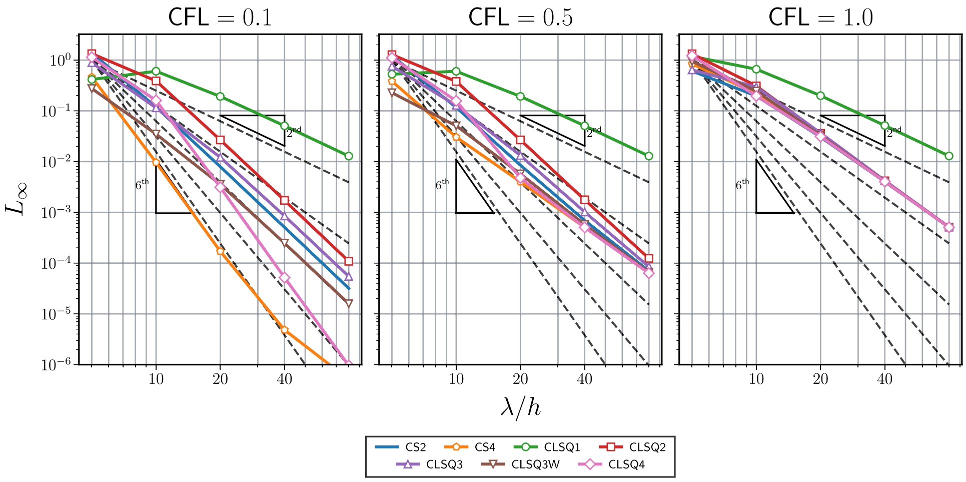
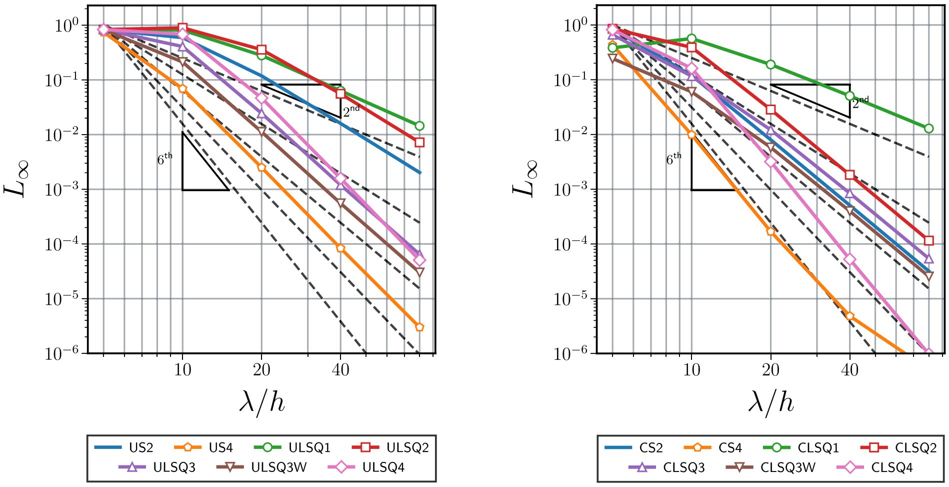


Fig. 13. L_∞ error norm of the solution in test case at $t = T$ using central schemes for three different values of the CFL and as a function of λ/h.



(a) Upwind schemes

(b) Central schemes

Fig. 14. L_∞ error norm of the solution in the two-dimensional linear advection test case with uniform meshes at $t = T$ using upwind (a) and central (b) schemes with $c\tau/h = 0.05$ and as a function of λ/h.

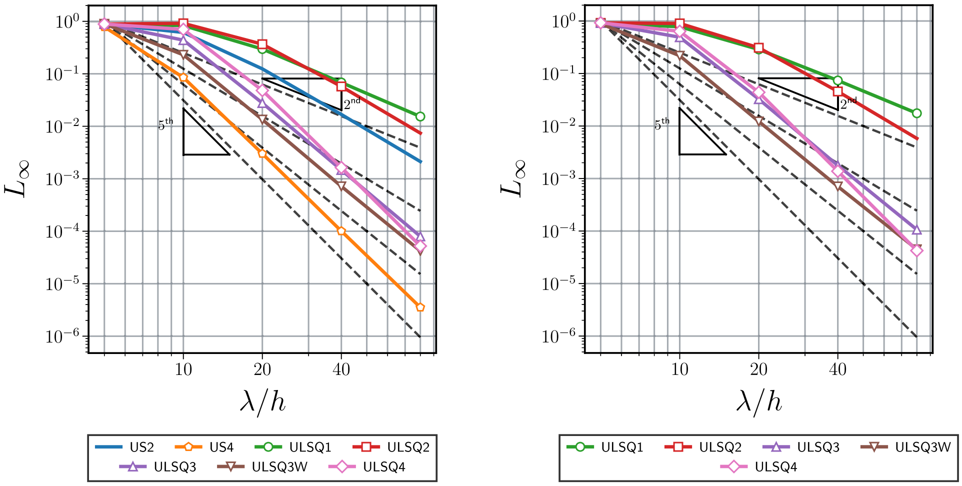
with the von Neumann analysis described in Section 5.1.2. Hence, the analysis of the aforementioned results is not included for the sake of brevity.

All the previous conclusions can also be applied to the error norms obtained using Mesh 2 and Mesh 3 configurations with the upwind schemes shown in Fig. 15a and Fig. 15b respectively. The results of central schemes are not presented as they are unstable in these mesh configurations. In these figures, it is possible to observe that LSQ schemes show similar error trends in non-uniform and unstructured meshes, maintaining the same order of accuracy as in structured meshes. This was also predicted in Section 5.2. S schemes results for Mesh 3 configuration are not presented due to the fact that they are only compatible with structured meshes.

To assess the behavior of the S schemes with meshes whose cell stencils principal directions present certain rotation with respect to those of the mesh, a new mesh configuration called Mesh 4 will be studied. This mesh is produced with the following transformation

$$\boldsymbol{x}' = \begin{cases} \boldsymbol{x} + \mathcal{K}_\theta L \begin{pmatrix} 0 \\ \sin\left(4\pi \frac{x}{L}\right) \end{pmatrix} + \mathcal{K} h \,\mathrm{rand}(-1,1) & \boldsymbol{x} \in (0,L)^2 \\ \boldsymbol{x} + \mathcal{K}_\theta L \begin{pmatrix} 0 \\ \sin\left(4\pi \frac{x}{L}\right) \end{pmatrix} & \boldsymbol{x} \notin (0,L)^2 \end{cases}, \tag{93}$$

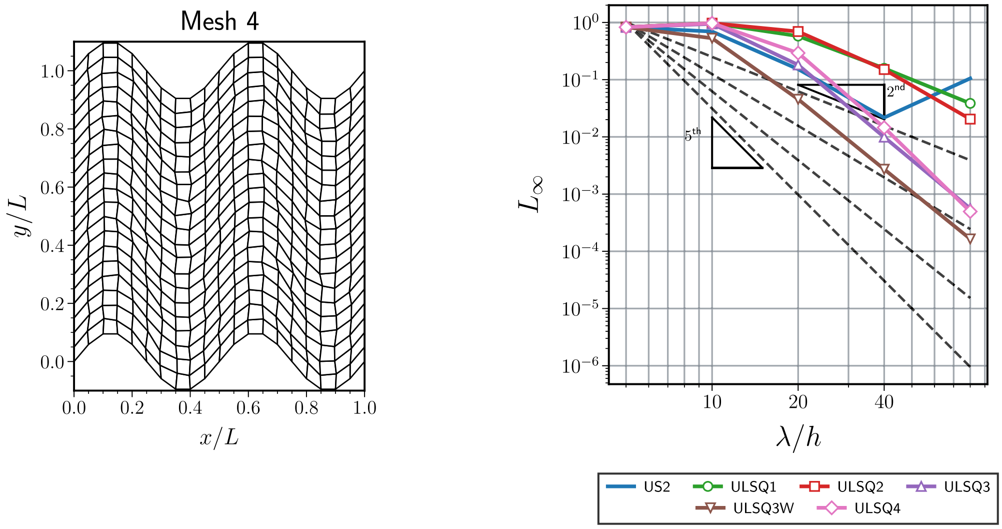
where $\mathcal{K} = \mathcal{K}_\theta = \frac{1}{10}$ for this mesh configuration. Fig. 16a shows this mesh configuration for $h = L/20$.



(a) Mesh 2

(b) Mesh 3

Fig. 15. L_∞ error norm of the solution in the two-dimensional linear advection test case with Mesh 2 (a) and 3 (b) configurations at $t = T$ using upwind schemes, $c\tau/h = 0.05$ and as a function of λ/h.



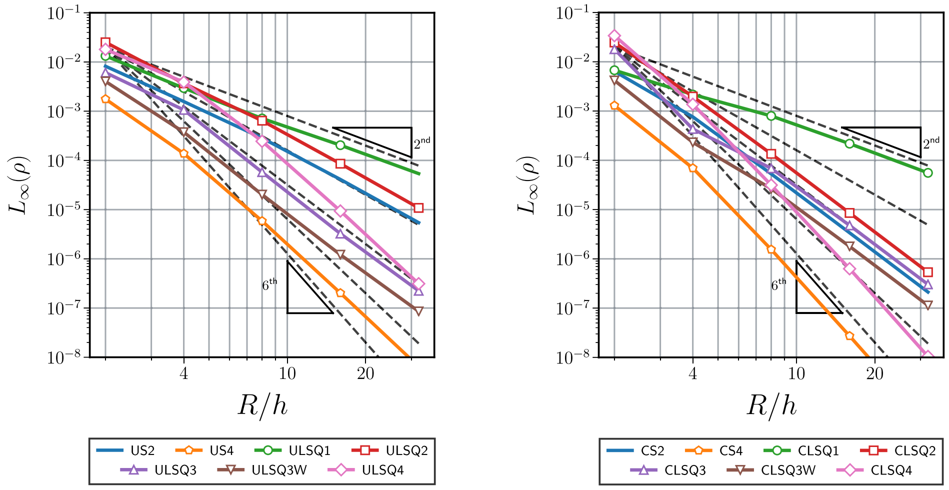
(a) Mesh 4 configuration

(b) L_∞ error norm

Fig. 16. L_∞ error norm of the solution in the two-dimensional linear advection test case with Mesh 4 configuration (a) at $t = T$ using upwind schemes, $c\tau/h = 0.05$ and as a function of λ/h.

Fig. 16b shows the error norm for the Mesh 4 configuration and the upwind central schemes respectively. It can be observed that the US2 scheme errors do not present a smooth behavior as the mesh is refined. In our observations, the increase of the error values is related to mesh regions with maximum curvature. This implies that the S schemes stability is sensitive to mesh curvature. Our own studies have shown that the $\mathcal{A}$ matrix of the S2 scheme becomes singular in mesh regions with a relative orientation to the $x - y$ principal axis close to 45°. Moreover, in this mesh configuration, the US4 schemes (results not shown in the figure) are only stable for the coarsest mesh, showing that the sensitivity of S schemes to mesh curvature increases with the polynomial degree of these schemes. At last, in this mesh configuration, the LSQ schemes present similar behavior to that reported in the previous analysis of Mesh 1, Mesh 2 and Mesh 3 configurations.

The performance of the S schemes could be improved in the Mesh 4 configuration by using a local reference frame for each stencil. However, due to the random perturbation added to each node of the mesh, the computation of a local reference frame is not straightforward. In our studies, we have tried several different reference frames described in [16] and the linear mapping proposed in Eq. (28). These coordinate transformations tended to increase numerical dissipation and dispersion and could not always yield stable solutions in the proposed structured meshes. Moreover, several non-linear mappings were also tested. With these transformations the schemes presented better stability properties although their



(a) Upwind schemes

(b) Central schemes

Fig. 17. L_∞ error norm of the density in the Euler Vortex test case with uniform meshes (Mesh 1) at $T = t_c$ for upwind (a) and central (b) schemes and as a function of R/h.

accuracy was degraded due to undetermined reasons. Further studies are required to generate local reference frames for S schemes to ensure accuracy and stability in arbitrary structured meshes.

6.4. Isentropic Euler-vortex in two-dimensional mesh

The two-dimensional static isentropic Euler vortex problem is commonly used to test the order of accuracy of numerical methods for conservation laws with non-linear fluxes [52], in particular for the Euler equations. The analytical solution of this problem in primitive variables is defined as follows:

$$\rho(x, y, t) = \left(1 - \frac{\gamma - 1}{2\gamma}\Gamma^2\right)^{\frac{1}{\gamma-1}} \quad , \tag{94}$$

$$\boldsymbol{u}(x, y, t) = \frac{\Gamma}{R}\begin{pmatrix} -y \\ x \end{pmatrix} \quad , \tag{95}$$

$$P(x, y, t) = \left(1 - \frac{\gamma - 1}{2\gamma}\Gamma^2\right)^{\frac{\gamma}{\gamma-1}} \quad , \tag{96}$$

where R is the radius of the vortex, and Γ is defined as

$$\Gamma = \Gamma_0 \mathrm{e}^{-\frac{1}{2\sigma^2}\left[\frac{x^2+y^2}{R^2}\right]} \quad . \tag{97}$$

In the previous expression Γ_0 and σ are constants which are set to 1. We define the characteristic time as $t_c = \frac{R^2}{\Gamma_0}$. The initial conditions of the conservative variables of the Euler equations are computed from the previously described primitive variables.

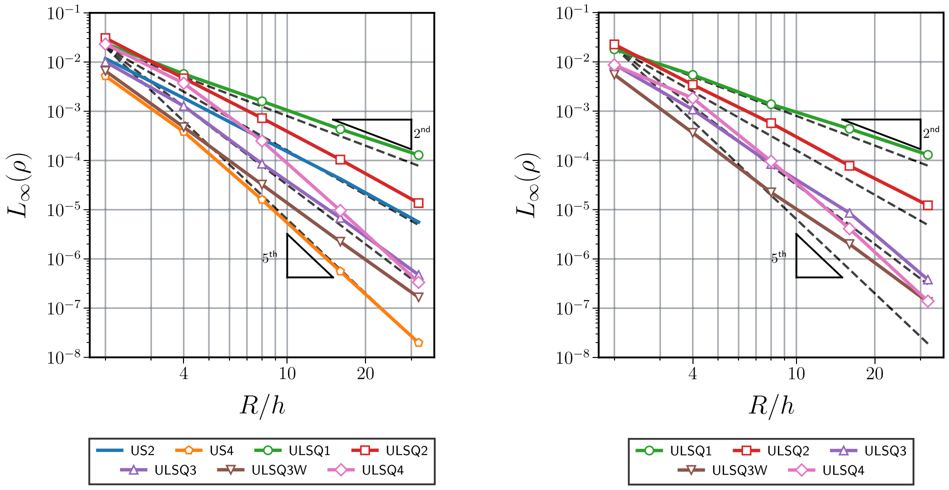
The computational domain is set to $\boldsymbol{x} \in [-L, L]^2$. Periodic boundary conditions are applied at each edge of the domain. It is worth noting that the choice of periodic boundary conditions can have an important influence on the order of accuracy analysis due to the non-periodic nature of the analytical and initial solutions. To avoid spurious numerical errors due to periodic boundary conditions, [52] stated that the length of the domain must be extended enough to ensure that the amplitude of the analytical solution at the edges of the domain is below or close to machine precision which is reached with $L/R = 10$ for double precision operations.

To reduce temporal discretization errors, the time step of the simulations is chosen as

$$\frac{\tau}{t_c} = \frac{1}{25}\frac{h}{R} \quad . \tag{98}$$

To analyze the accuracy of the different schemes, the L_∞ error norm of the density as a function of the number of cells per radius of the vortex R/h is considered at physical time $T = t_c$ for the different schemes and Mesh 1, 2 and 3 configurations.

Fig. 17a, 18a shows the results of the upwind schemes in Mesh 1 and Mesh 2 configuration respectively. These results are similar to those obtained in the one and two-dimensional linear advection analysis. The ULSQ3W outperforms all LSQ



(a) Mesh 2

(b) Mesh 3

Fig. 18. L_∞ error norm of the density in the Euler Vortex test case with Mesh 2 (a) and Mesh 3 (b) configuration at $T = t_c$ for different upwind schemes and as a function of R/h. Note that the S schemes are missing as they are not compatible with unstructured meshes.

schemes for the studied range of R/h and the S schemes show lower errors than those of their LSQ counterparts. All schemes show the predicted order of accuracy. Similar conclusions can be obtained in Mesh 3 configuration as shown in Fig. 18b, although in this mesh configuration the S schemes cannot be used due to the unstructured properties of the mesh.

Fig. 17b shows the error values of the central schemes in the static Euler Vortex test case. As for upwind schemes, there results are similar to those showed in the one and two-dimensional linear advection analysis. CLSQ3W scheme shows less error for under-resolved simulations while CLSQ4 presents the lowest error values of LSQ schemes for well-resolved meshes. S schemes outperform all their LSQ counterparts. All central schemes built with an even degree polynomial show k+1-exactness. An important conclusion that can be obtained from these results is the fact that the analytical findings regarding the order of accuracy of finite-volume schemes in the linear-advection equation for uniform meshes can also be applied for this non-linear problem when using the analyzed schemes (in particular the high-order face average fluxes reconstruction). For even degree polynomial reconstruction, this conclusion cannot be straightforwardly drawn from our analytical findings, as for non-linear problems it is not generally possible to obtain an analytical expression of the truncation error.

6.5. Navier-Stokes decaying compressible Homogeneous Isotropic Turbulence (HIT)

To assess the performance of the studied numerical schemes in non-linear problems in which there exists an interaction between different wavenumbers or Fourier modes, the analysis of a test case of decaying Compressible Homogeneous Turbulence in the Navier-Stokes equations is proposed [53–57].

The simulation is performed in a three-dimensional box such that $\boldsymbol{x} \in [0, 2\pi]^3$ with periodic boundary conditions on all the boundaries of the domain containing 64^3 cells. The mesh used to discretize the domain is structured and uniform. The density and pressure fields are initialized with constant values ρ_0 and P_0 while the initial velocity field, denoted as $\boldsymbol{u}_0$, corresponds to a random solenoidal vector field whose energy spectrum is given by

$$E(\kappa) = 16 u_{\mathrm{RMS},0}^2 \sqrt{\frac{2}{\pi}} \frac{\kappa^4}{\kappa_0^5} \mathrm{e}^{-2\frac{\kappa^2}{\kappa_0^2}} \quad , \tag{99}$$

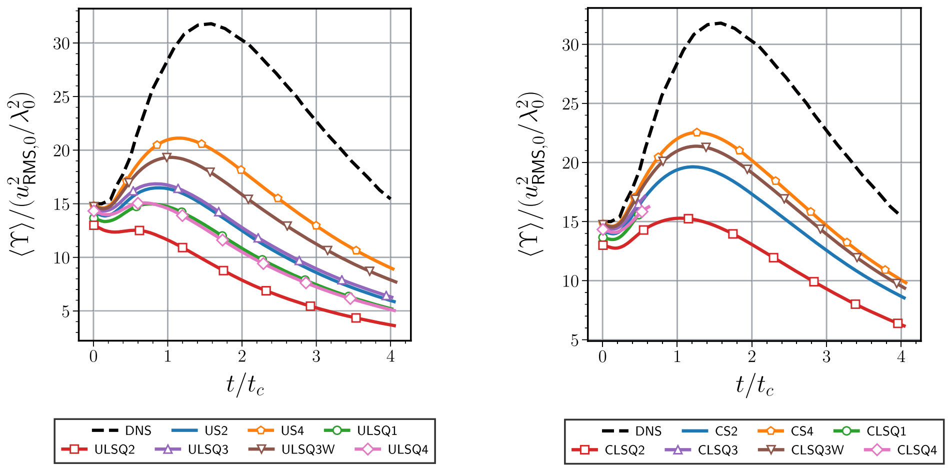
where $\kappa_0 = 4$ and $u_{\mathrm{RMS},0}$ is the root mean squared value of the initial velocity field. The value of the latter variable is related to the initial energy spectrum as

$$\frac{3}{2} u_{\mathrm{RMS},0}^2 = \frac{\langle \boldsymbol{u}_0 \cdot \boldsymbol{u}_0 \rangle}{2} = \int_{\kappa=0}^{\infty} E(\kappa)\, \mathrm{d}\kappa \quad , \tag{100}$$

where $\langle \square \rangle$ symbolizes an ensemble average over all the domain

$$\langle \square \rangle = \frac{1}{|\Omega|} \int_{\Omega} \square \, \mathrm{d}\Omega \quad . \tag{101}$$

The root mean squared value of the initial velocity field can be computed from the following simulation parameters from [54] and [56]



(a) Upwind schemes

(b) Central schemes

Fig. 19. Non-dimensional enstrophy temporal evolution in the decaying compressible HIT test case with $M_{t,0} = 0.6$ for upwind (a) and central (b) schemes. DNS data digitized from [56].

$$Re_{\lambda_0} = \frac{\rho_0 u_{\mathrm{RMS},0}\lambda_0}{\mu} = 100 \quad , \quad M_{t,0} = \frac{\sqrt{\langle \boldsymbol{u}_0 \cdot \boldsymbol{u}_0\rangle}}{\sqrt{\gamma \frac{p_0}{\rho_0}}} = 0.6 \qquad , \tag{102}$$

and λ_0 refers to the initial length of the Taylor's scales in this simulation. Parameters Re_{λ_0} and $M_{t,0}$ are the initial Reynolds number based on the Taylor scales and the initial turbulent Mach number respectively. The viscosity μ is considered as a constant following [56]. Please note that the choice of energy spectrum implies $\lambda_0 = \frac{2}{\kappa_0}$.

The initial velocity field is generated following the procedure described in [54] using a tool provided by the suite PeleC [56]. As the analysis of this section is not focused on the asymptotic behavior of the numerical error, no distinction between cell averages and point values will be made in the initialization process.

The temporal evolution of the initial prescribed velocity spectrum generates an energy cascade which produces energy exchanges between larger and the small dissipative structures. This implies that the numerical discetization errors of under-resolved waves have an impact on the global numerical errors of the simulations. This test case differs drastically from the non-linear isentropic Euler Vortex as in the latter there exists an analytical solution which shows that there is no interaction between different wavenumbers of the Fourier modes of the solution. Nonetheless, there is no guarantee that the numerical solution of the Euler Vortex doesn't present interaction between different wavenumbers, although the error analysis performed shows that this interaction doesn't have an important influence in the solution as the mesh resolution increases, at least for the configuration tested in this work. Therefore, this study should allow to determine the influence of the numerical errors due to under-resolved waves in the prediction of turbulence flows.

At last, as a result of the chosen turbulent Mach number, the simulations performed using central schemes tend to be unstable due to the appearance of eddy shocklets [54,57]. Therefore, to stabilize the simulations using central schemes, the shock sensor described [58] is used to modify the value of f_g at each quadrature point g such that

$$f_g = \left\{ \begin{matrix} 1 & \text{if} & \frac{-\nabla\cdot\boldsymbol{u}_g}{\nabla\cdot\boldsymbol{u}_g + \|\nabla\times\boldsymbol{u}_g\|} \geq 0.65 \\ 0 & \text{if} & \frac{-\nabla\cdot\boldsymbol{u}_g}{\nabla\cdot\boldsymbol{u}_g + \|\nabla\times\boldsymbol{u}_g\|} < 0.65 \end{matrix} \right\} \quad . \tag{103}$$

It is worth mentioning, that central schemes could be stabilized using other limiters, shock sensors, schemes, etc. Interested reader is referred to the works of [54,57] for more information on this topic.

Fig. 19a shows the temporal evolution of the enstrophy ensemble average with upwind schemes computed as

$$\Upsilon = \langle \|\nabla \times \boldsymbol{u}\|^2 \rangle \quad . \tag{104}$$

We note that the post-processing of the results of each scheme is performed using the velocity gradient of the studied scheme. Due to the different precisions of schemes regarding the computation of this gradient, the initial values of variables which depend on the velocity gradient (such as the enstrophy) differ between schemes. The non-dimensionalization factor could be taken as the initial numerical value of the variable considered. However, this option was not chosen in this work because it would mask the disparity in accuracy of the gradient reconstruction between the different schemes. For comparison purposes, reference DNS data from [56] obtained with very-high-order finite-different method code SMC [59] and 512^3 points is added to the comparison. The characteristic time $t_c = \frac{\lambda_0}{u_{\mathrm{RMS},0}}$ is used to normalize the temporal evolution.

The results show that all schemes under-predict the production of enstrophy by a large margin in the studied mesh. If the schemes are sorted according to their error with respect to the reference DNS data, it is possible to observe that their ordering closely resembles the one obtained from the errors shown in figure Fig. 17a for under-resolved simulations. It can be observed that the results from ULSQ4 are far less accurate than those of the US4 scheme although both have the same order of accuracy and similar stencil sizes. These conclusions are also obtained when comparing ULSQ2 and USLQ3 with US2 schemes. Moreover, the use of least-squares weighting in the ULSQ3W scheme significantly improves the accuracy compared to its counterpart ULSQ3 scheme, yielding the closest results to the US4 scheme while using a substantially smaller stencil. These results seem to indicate that this simulation is drastically under-resolved although it is representative of real ILES turbulent flow test cases in which the resolution of all scales is not computationally possible. One conclusion to be obtained from these results is that the asymptotic behavior of the studied schemes cannot be directly correlated with the expected accuracy of the schemes in under-resolved turbulent flow cases.

Fig. 19b shows the temporal evolution of the enstrophy ensemble average with central schemes using the shock sensor described in Eq. (103). All central LSQ schemes except the CLSQ3W scheme showed unstable results, either by divergence of the simulation or by over-predicting the total kinetic energy of the flow (this variable is not shown in this section for the sake of brevity). To ensure that these unstable results are not related to the use of a shock sensor, the same simulations were performed without shock sensor and using a lower Mach number $M = 0.2$ and the unstable CLSQ schemes (this results from these simulations are not shown for the sake of brevity). In the latter simulations there was no divergence of the solution, although all the LSQ schemes showed an important over-prediction of the temporal evolution of the total kinetic energy. This implies that the behavior of the default CLSQ schemes tends to be unstable in non-linear under-resolved turbulent flow cases. The addition of LSQ weighting distance methods as in the CLSQ3W scheme drastically improves the numerical stability compared to the default CLSQ3 scheme. At last, it is important to note the improvement in the prediction of the enstrophy when using central schemes compared to the results of upwind schemes. Nonetheless, these improvements are partially limited due to the use of shock sensors. One should expect even more differences between central and upwind schemes for lower Mach number cases.

7. Conclusions

The numerical properties of the finite volume least-squares spatial discretization schemes have been studied and compared to finite-volume schemes especially catered for structured meshes. The k-exactness properties of all schemes in uniform meshes have been investigated through one-dimensional and two-dimensional analysis of the truncation error. A novel method has been developed to assess the k-exact properties of the schemes in all kind of meshes when dealing with the discretization of conservative laws through the finite volume method. This method shows that the order of accuracy of some of these schemes can be improved by using central reconstructions, although the use of central schemes in non-uniform meshes yields unstable simulations. Moreover, an extensive analysis of their dissipation and dispersion properties in uniform and unstructured meshes has been performed, extending the von Neumann's formulation to certain non-uniform and unstructured meshes. This analysis showed that, in uniform one-dimensional and two-dimensional meshes, the least-squares schemes have worse dissipation and dispersion properties than other spatial discretization schemes developed for structured uniform meshes, specially in the under-resolved part of the spectrum. Moreover, the dissipation and dispersion study in two-dimensional non-uniform meshes proved that the dissipation and dispersion properties of the least-squares schemes are not heavily influenced by the non-uniformity of the studied meshes of this work. Furthermore, we have theoretically and numerically demonstrated that the addition of weighting distance to the least-squares schemes can improve the dissipation and dispersion properties of these schemes. To validate the analytical findings, several one-dimensional and two-dimensional simulations have been performed with linear and non-linear equations and with uniform and non-uniform meshes proving that the analytical findings can also be observed in the simulations. Special emphasis is placed on the behavior of these LSQ schemes in under-resolved simulations of turbulent flows. In these cases, it has been observed that the default LSQ schemes drastically under-perform compared to schemes for structured meshes, although the addition of weighting distance methods to the LSQ formulation notably improves the results. Moreover, the analysis showed that the numerical performance of the schemes follows closely that obtained in less complex non-linear simulations. This demonstrates that the asymptotic analysis of the numerical schemes is not always representative of the behavior of the numerical schemes in coarse turbulent test cases. To conclude, it has been shown that least-squares finite volume schemes allow to easily obtain finite volume k-exact schemes for arbitrary quadrangular meshes in linear and non-linear cases. However, this comes at a cost, the dissipation and dispersion properties of these schemes are not optimized for good-quality meshes and/or under-resolved simulations, which is critical for ILES simulations in which the important flow regions are usually discretized using near-to-uniform meshes and in which not all scales can be correctly captured due to the prohibitively associated computational cost. The addition of weighting distance methods to the LSQ formulation has been shown to greatly improve the numerical performance of the LSQ schemes. Further studies are been carried out to improve the numerical properties of least-squares schemes in near-to uniform meshes.

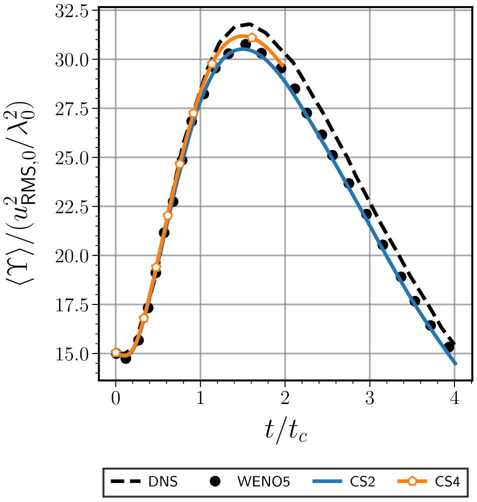


Fig. A.20. Non-dimensional enstrophy temporal evolution in the decaying compressible HIT test case with $M_{t,0} = 0.6$ using S central schemes together with the shock sensor described in Eq. (103). Results obtained with 256^3 cells. DNS data digitized from [56]. WENO5 (5th order) data was obtained using 256^3 and it was digitized from [56].

CRediT authorship contribution statement

G. Saez-Mischlich: Conceptualization, Methodology, Software, Validation, Writing – original draft. **J. Sierra-Ausin:** Conceptualization, Methodology, Writing – original draft. **G. Grondin:** Supervision, Writing – review & editing. **J. Gressier:** Supervision, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

This work was performed using HPC resources from GENCI-IDRIS and GENCI-CINES on Jean Zay, Occigen (Grant A0082A07178) and CALMIP on Olympe (Grant 2020-p1425).

Appendix A. HIT initial condition validation

To ensure that the initial condition to simulate the HIT test case yields results which can be compared with DNS reference data of [56], Fig. A.20 shows the ensemble enstrophy as a function of the time for the central S schemes in a 256^3 mesh. The results are compared with DNS and FV-WENO5 (5th order scheme) data that obtained from [56]. FV-WENO5 was obtained using the same mesh discretization (256^3 cells). This FV-WENO5 scheme is a finite-volume scheme which uses the S4 scheme formulation described in this work (Section 3.2) in combination with a WENO method to better treat solution discontinuities. The results show almost good agreement with DNS data and that the CS4 scheme with the Ducros shock sensor yields more accurate results than those of the FV-WENO5 scheme. This figure also serves as a validation for both the initial condition of the HIT test case and the implementation of the high-order viscous fluxes discretization described in Section 3.4 and Section 2.3.

References

[1] H. Liu, X. Jiao, Wls-eno: weighted-least-squares based essentially non-oscillatory schemes for finite volume methods on unstructured meshes, J. Comput. Phys. 314 (2016) 749–773.
[2] P. Tsoutsanis, V. Titarev, D. Drikakis, Weno schemes on arbitrary mixed-element unstructured meshes in three space dimensions, J. Comput. Phys. 230 (2011) 1585–1601.
[3] P. Tsoutsanis, Stencil selection algorithms for weno schemes on unstructured meshes, J. Comput. Phys. X 4 (2019) 100037.
[4] P.S. Farmakis, P. Tsoutsanis, X. Nogueira, Weno schemes on unstructured meshes using a relaxed a posteriori mood limiting approach, Comput. Methods Appl. Mech. Eng. 363 (2020) 112921.
[5] T. Martin, I. Shevchuk, Implementation and validation of semi-implicit weno schemes using openfoam®, Computation 6 (2018) 6.
[6] C. Ollivier-Gooch, M.V. Altena, A high-order-accurate unstructured mesh finite-volume scheme for the advection–diffusion equation, J. Comput. Phys. 181 (2002) 729–752.

[7] M. Dumbser, M. Käser, Arbitrary high order non-oscillatory finite volume schemes on unstructured meshes for linear hyperbolic systems, J. Comput. Phys. 221 (2007) 693–723.
[8] R. Zhang, M. Zhang, C.W. Shu, On the order of accuracy and numerical performance of two classes of finite volume WENO schemes, Commun. Comput. Phys. (2011).
[9] V. Titarev, E. Toro, Finite-volume weno schemes for three-dimensional conservation laws, J. Comput. Phys. 201 (2004) 238–260.
[10] F. Haider, B. Courbet, J.-P. Croisille, A high-order interpolation for the finite volume method: the coupled least squares reconstruction, Comput. Fluids 176 (2018) 20–39.
[11] P. Tsoutsanis, I.W. Kokkinakis, L. Könözsy, D. Drikakis, R.J. Williams, D.L. Youngs, Comparison of structured- and unstructured-grid, compressible and incompressible methods using the vortex pairing problem, Comput. Methods Appl. Mech. Eng. 293 (2015) 207–231.
[12] P. McCorquodale, P. Colella, A high-order finite-volume method for conservation laws on locally refined grids, Commun. Appl. Math. Comput. Sci. 6 (2011) 1–25.
[13] Y. Du, A high-order modified finite volume WENO method on 3d Cartesian grids, Commun. Comput. Phys. 26 (2019) 768–784.
[14] G. Pont, P. Brenner, P. Cinnella, B. Maugars, J.-C. Robinet, Multiple-correction hybrid k-exact schemes for high-order compressible rans-les simulations on fully unstructured grids, J. Comput. Phys. 350 (2017) 45–83.
[15] T. Arbogast, C.-S. Huang, X. Zhao, Finite volume weno schemes for nonlinear parabolic problems with degenerate diffusion on non-uniform meshes, J. Comput. Phys. 399 (2019) 108921.
[16] A. Fosso Pouangué, H. Deniau, F. Sicot, P. Sagaut, Curvilinear finite-volume schemes using high-order compact interpolation, J. Comput. Phys. 229 (2010) 5090–5122.
[17] A. Jalali, C.F.O. Gooch, Higher-order finite volume solution reconstruction on highly anisotropic meshes, in: 21st AIAA Computational Fluid Dynamics Conference, American Institute of Aeronautics and Astronautics, 2013.
[18] A.G. Vasconcelos, D.M. Albuquerque, J.C. Pereira, A very high-order finite volume method based on weighted least squares for elliptic operators on polyhedral unstructured grids, Comput. Fluids 181 (2019) 383–402.
[19] M. Feistauer, M. Feistauer, J. Felcman, J. Felcman, I. Straškraba, Mathematical and Computational Methods for Compressible Flow, Oxford University Press on Demand, 2003.
[20] M. Ndjinga, Spectral stability of finite volume schemes for linear hyperbolic systems, C. R. Math. 349 (2011) 1111–1115.
[21] R. Naff, T. Russell, J. Wilson, Shape functions for velocity interpolation in general hexahedral cells, Comput. Geosci. 6 (2002) 285–314.
[22] J.C. Butcher, Numerical Methods for Ordinary Differential Equations, John Wiley & Sons, Ltd., 2008.
[23] P. Tsoutsanis, Extended bounds limiter for high-order finite-volume schemes on unstructured meshes, J. Comput. Phys. 362 (2018) 69–94.
[24] P. Roe, Approximate Riemann solvers, parameter vectors, and difference schemes, J. Comput. Phys. 43 (1981) 357–372.
[25] E.F. Toro, The HLL and HLLC Riemann solvers, in: Riemann Solvers and Numerical Methods for Fluid Dynamics, Springer, Berlin Heidelberg, 1999, pp. 315–339.
[26] G. Gassner, F. Lörcher, C.-D. Munz, A contribution to the construction of diffusion fluxes for finite volume and discontinuous Galerkin schemes, J. Comput. Phys. 224 (2007) 1049–1063.
[27] A. Jalali, M. Sharbatdar, C. Ollivier-Gooch, Accuracy analysis of unstructured finite volume discretization schemes for diffusive fluxes, Comput. Fluids 101 (2014) 220–232.
[28] Q. Wang, Y.-X. Ren, W. Li, Compact high order finite volume method on unstructured grids i: basic formulations and one-dimensional schemes, J. Comput. Phys. 314 (2016) 863–882.
[29] L. Cueto-Felgueroso, I. Colominas, X. Nogueira, F. Navarrina, M. Casteleiro, Finite volume solvers and moving least-squares approximations for the compressible Navier-Stokes equations on unstructured grids, Comput. Methods Appl. Mech. Eng. 196 (2007) 4712–4736.
[30] H. Xiao, Z. Gimbutas, A numerical algorithm for the construction of efficient quadrature rules in two and higher dimensions, Comput. Math. Appl. 59 (2010) 663–676.
[31] J. Jaśkowiec, N. Sukumar, High-order cubature rules for tetrahedra, Int. J. Numer. Methods Eng. 121 (2020) 2418–2436.
[32] N. Schlömer, N. Papior, D. Arnold, M. Ancellin, R. Zetter, nschloe/quadpy v0.16.4, https://doi.org/10.5281/zenodo.4288881, 2020.
[33] Y. Abe, T. Haga, T. Nonomura, K. Fujii, On the freestream preservation of high-order conservative flux-reconstruction schemes, J. Comput. Phys. 281 (2015) 28–54.
[34] C.L. Lawson, J. Richard Hanson, S. for Industrial A. Mathematics, Solving Least Squares Problems, rev ed., SIAM, Philadelphia, 1995, http://www.loc.gov/catdir/enhancements/fy0617/95035178-t.html, "This SIAM edition is an unabridged, revised republication of the work first published by Prentice-Hall, Inc., Englewood Cliffs, New Jersey, 1974"–T.p. verso.
[35] G. Guennebaud, B. Jacob, et al., Eigen v3, http://eigen.tuxfamily.org, 2010.
[36] N. Petrovskaya, The accuracy of least-squares approximation on highly stretched meshes, Int. J. Comput. Methods 5 (2008) 449–462.
[37] C.-W. Shu, Essentially non-oscillatory and weighted essentially non-oscillatory schemes for hyperbolic conservation laws, Springer Berlin Heidelberg, Berlin, Heidelberg, 1998, pp. 325–432.
[38] J. Epperson, An Introduction to Numerical Methods and Analysis, John Wiley & Sons, Inc., Hoboken, New Jersey, 2013.
[39] R. Burden, Numerical Analysis, Brooks/Cole, Australia, Pacific Grove, CA, 2001.
[40] J. Manzanero, G. Rubio, E. Ferrer, E. Valero, D.A. Kopriva, Insights on aliasing driven instabilities for advection equations with application to Gauss–Lobatto discontinuous Galerkin methods, J. Sci. Comput. 75 (2017) 1262–1281.
[41] L. Reichel, L.N. Trefethen, Eigenvalues and pseudo-eigenvalues of Toeplitz matrices, Linear Algebra Appl. 162 (1992) 153–185.
[42] S.K. Lele, Compact finite difference schemes with spectral-like resolution, J. Comput. Phys. 103 (1992) 16–42.
[43] P. Vincent, P. Castonguay, A. Jameson, Insights from von Neumann analysis of high-order flux reconstruction schemes, J. Comput. Phys. 230 (2011) 8134–8154.
[44] B. Vermeire, P. Vincent, On the properties of energy stable flux reconstruction schemes for implicit large Eddy simulation, J. Comput. Phys. 327 (2016) 368–388.
[45] M. Alhawwary, Z. Wang, Fourier analysis and evaluation of dg, fd and compact difference methods for conservation laws, J. Comput. Phys. 373 (2018) 835–862.
[46] W. Trojak, R. Watson, A. Scillitoe, P.G. Tucker, Effect of mesh quality on flux reconstruction in multi-dimensions, J. Sci. Comput. 82 (2020).
[47] J. Vanharen, G. Puigt, X. Vasseur, J.-F. Boussuge, P. Sagaut, Revisiting the spectral analysis for high-order spectral discontinuous methods, J. Comput. Phys. 337 (2017) 379–402.
[48] T.E. Voth, M.J. Martinez, M.A. Christon, Generalized Fourier analyses of the advection–diffusion equation—part II: two-dimensional domains, Int. J. Numer. Methods Fluids 45 (2004) 889–920.
[49] R. Vichnevetsky, J.B. Bowles, Fourier Analysis of Numerical Approximations of Hyperbolic Equations, Society for Industrial and Applied Mathematics, 1982.
[50] P. Fernandez, R.C. Moura, G. Mengaldo, J. Peraire, Non-modal analysis of spectral element methods: towards accurate and robust large-Eddy simulations, Comput. Methods Appl. Mech. Eng. 346 (2019) 43–62.
[51] G. Mengaldo, D.D. Grazia, P.E. Vincent, S.J. Sherwin, On the connections between discontinuous Galerkin and flux reconstruction schemes: extension to curvilinear meshes, J. Sci. Comput. 67 (2015) 1272–1292.

[52] S.C. Spiegel, H. Huynh, J.R. DeBonis, A survey of the isentropic Euler vortex problem using high-order methods, in: 22nd AIAA Computational Fluid Dynamics Conference, American Institute of Aeronautics and Astronautics, 2015.
[53] S. Lee, S.K. Lele, P. Moin, Eddy shocklets in decaying compressible turbulence, Phys. Fluids A, Fluid Dyn. 3 (1991) 657–664.
[54] E. Johnsen, J. Larsson, A.V. Bhagatwala, W.H. Cabot, P. Moin, B.J. Olson, P.S. Rawat, S.K. Shankar, B. Sjögreen, H. Yee, X. Zhong, S.K. Lele, Assessment of high-resolution methods for numerical simulations of compressible turbulence with shock waves, J. Comput. Phys. 229 (2010) 1213–1237.
[55] D. Kotov, H. Yee, A. Wray, B. Sjögreen, A. Kritsuk, Numerical dissipation control in high order shock-capturing schemes for LES of low speed flows, J. Comput. Phys. 307 (2016) 189–202.
[56] E. Motheau, J. Wakefield, Investigation of finite-volume methods to capture shocks and turbulence spectra in compressible flows, Commun. Appl. Math. Comput. Sci. 15 (2020) 1–36.
[57] J. Fernández-Fidalgo, L. Ramírez, P. Tsoutsanis, I. Colominas, X. Nogueira, A reduced-dissipation WENO scheme with automatic dissipation adjustment, J. Comput. Phys. (2020) 109749.
[58] J. Larsson, S. Lele, P. Moin, Effect of numerical dissipation on the predicted spectra for compressible turbulence, in: Annual Research Briefs 2007, Center for Turbulence Research, 2007.
[59] M. Emmett, W. Zhang, J.B. Bell, High-order algorithms for compressible reacting flow with complex chemistry, Combust. Theory Model. 18 (2014) 361–387.