Journal of Computational Physics 524 (2025) 113735

Contents lists available at ScienceDirect

Journal of Computational Physics

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

Order-lifted data inversion/retrieval method of neighbor cells to implement general high-order schemes in unstructured-mesh-based finite-volume solution framework

Hao Guo, Boxing Hu, Peixue Jiang, Xiaofeng Ma, Yinhai Zhu *

Key Laboratory for Thermal Science and Power Engineering of Ministry of Education, Key Laboratory of CO_2 Utilization and Reduction Technology, Department of Energy and Power Engineering, Tsinghua University, Beijing, 100084, China

ARTICLE INFO

Keywords:
Unstructured gird
High-order method
Finite-volume method

ABSTRACT

This study introduces an order-lifted inversion/retrieval method for implementing high-order schemes within the framework of an unstructured-mesh-based finite-volume method. This method defines a special representation called the data order-lifted inversion of neighbor cells (DOLINC) differential, which transforms the degrees of freedom of wide templates into differentials of various orders stored in local grid cells. Furthermore, to retrieve the original far-field information without bias during the reconstruction/interpolation of face values, the corresponding accurate inversion formulas are derived based on the defined DOLINC differentials. The order-lifted inversion method can be applied to multi-dimensional polyhedral-mesh solvers by considering the influence of grid non-uniformity on high-order schemes. It seamlessly accommodates multi-process parallel computing for high-order methods without requiring special consideration for the boundary interface. This method not only enhances the numerical accuracy of second-order finite-volume methods, but also demonstrates a significant computational-speed advantage over similar methods. A series of benchmark cases, including the linear advection, Burgers, and Euler equations, are comprehensively validated to assess the practical performance of the method. The results indicate that the unstructured-mesh high-order schemes implemented based on this method achieve theoretical accuracy in practical computations and substantially reduce computational costs compared with methods that increase grid resolution.

1. Introduction

Recently, high-order computational fluid dynamics (CFD) methods have gained widespread recognition in the academic community. These methods, which are known for capturing more physical details with fewer grid points, can significantly reduce the computational costs of complex problems. Various fields that involve fluid-simulation problems, such as aerospace, energy, environmental sciences, and biomedical sciences, urgently need more precise simulation methods. Among these, rapidly evolving scale-resolving turbulence-simulation methods such as the detached eddy simulation, large eddy simulation (LES), and direct numerical simulation (DNS) have been increasingly applied to various complex flow problems [8,18,43]. Increasing the order of the numerical schemes is highly efficient for obtaining higher accuracy in capturing the flow details. For example, when the desired char-

* Corresponding author.
E-mail address: yinhai.zhu@tsinghua.edu.cn (Y. Zhu).

https://doi.org/10.1016/j.jcp.2025.113735
Received 14 April 2024; Received in revised form 15 November 2024; Accepted 7 January 2025

Available online 10 January 2025
0021-9991/© 2025 Elsevier Inc. All rights are reserved, including those for text and data mining, AI training, and similar technologies.

acteristic length reaches 0.001, a second-order scheme requires 100 grid points, whereas a sixth-order scheme only needs 5.71 grid points [47]. This computational-efficiency improvement is particularly attractive for scale-resolving turbulence simulations, which are still constrained by computational costs, making them a focal point in related research.

Research on high-order schemes has been ongoing for nearly 40 years. Since the introduction of the concept of total variation diminishing (TVD) schemes by Harten [22], most high-order scheme research has focused on structured grids. However, CFD has been increasingly applied to complex flow problems; thus, unstructured grids, with their ability to handle intricate geometries and higher computational efficiency, are now widely adopted by most industrial software. The advantage of using an unstructured mesh is the substantial reduction in the required grid information storage. An unstructured mesh can be applied not only to polyhedron meshes of arbitrary shapes, but also to large-problem high-performance computing (HPC) for solving complex problems. However, unstructured grids encounter greater challenges in indirectly obtaining far-field-cell degree-of-freedom (DOF) information compared to structured grids, thereby presenting significant difficulties in implementing high-order schemes.

Barth and Frederickson initially proposed a k-exact reconstruction method to implement high-order schemes on unstructured grids [6]. The fundamental idea was to utilize a local template comprising multiple mesh cells and Gaussian integration points to construct a k-degree interpolation polynomial in the local space, thereby obtaining the corresponding $k+1$ order method. This method later became the foundation for many high-order methods for unstructured meshes [13,49], including the essentially non-oscillatory (ENO) and weighted essentially non-oscillatory (WENO) methods. The original ENO method was developed based on structured grids, and was first introduced by Harten et al. [23]. The basic concept involved selecting the smoothest reconstruction from several alternative templates when constructing the scheme. Durlofsky et al. [12] and Abgrall [1] subsequently extended the ENO method to unstructured grids. Similarly, based on an unstructured-mesh k-exact framework, Ollivier-Gooch [35] established a quasi-ENO scheme using least-squares reconstruction. The WENO method was initially proposed for structured grids by Liu et al. [33] and then by Jiang and Shu [28]. Friedrich [17] extended the WENO method to unstructured grids. Hu and Shu [24] obtained the WENO method for unstructured grids based on a two-dimensional triangular mesh. Dumbser et al. established WENO methods of arbitrary order based on an unstructured finite-volume method (FVM) framework [10] and subsequent Gauss-integration-free WENO methods [11]. Li and Ren [32] proposed an improved WENO method for unstructured grids under a k-exact framework using quadratic reconstruction. Tsoutsanis et al. implemented WENO methods for unsteady [46] and viscous flows [45] based on unstructured hybrid grids. Following Dumbser and Tsoutsanis' methods, Martin and Shevchuk [34] implemented WENO methods on the open-source C++ library OpenFOAM [14] using the unstructured-mesh-based FVM framework, and subsequent improvements in computational efficiency were made by Gärtner et al. [21]. Although these high-order methods, which were implemented based on k-exact methods, achieved theoretical accuracy on unstructured grids, they did not outperform or persist with high-order schemes on structured grids in terms of computational efficiency. This is because even without increasing the Gaussian integration points, k-exact high-order schemes still require local searches for multiple-cell templates and the calculation and storage of a large number of interpolation polynomial coefficients. For example, for a three-dimensional problem, as many as 19 template coefficients of $k = 3$ need to be stored. Most CFD industrial software, whether commercial codes, such as ANSYS Fluent [2] and STAR-CCM+ [41], or open-source codes, such as OpenFOAM and code_saturne [3], are based almost entirely on second-order FVM frameworks. Implementing k-exact high-order schemes on such low-order FVMs involves the challenge of increasing Gaussian integration points as well as the special handling of template-information communication at interfaces during parallel computing, which is unfavorable for both computational cost and stability.

Currently, one of the most prominent and highly promising high-order methods is the correction procedure via reconstruction (CPR) series. In 2007, Huynh [25] first introduced the flux reconstruction (FR) approach, unifying the discontinuous Galerkin, spectral volume, and spectral difference methods within a single framework. In 2009, Huynh [26] extended the FR method to two-dimensional quadrilateral grids. Wang proposed a lifting collocation penalty formulation in 2009, further generalizing the FR approach to two-dimensional triangular grids and triangular/quadrilateral hybrid grids [50]. Subsequently, Huynh [27] and Wang et al. [48] collectively termed these methods as CPR. Although CPR methods exhibit several excellent characteristics, the differences between CPR and the classical second-order FVM framework are even more pronounced compared to k-exact methods. Implementing the CPR method using most of the existing unstructured-grid FVM industrial software is almost impossible with top-level modifications. To utilize CPR methods, a bottom-up compatibility between the high-order mesh and corresponding solution framework must be rebuilt. Therefore, current applications of CPR methods are primarily implemented in in-house codes developed by researchers specializing in high-order scheme development rather than in widely used industrial software for general applications.

Therefore, although academic research has long developed high-order methods, and various high-order methods have been applied by scholars in different fields to achieve high-precision numerical simulations of complex problems, low-order FVM software still dominates current industrial applications. The widespread adoption of unstructured high-order schemes in industrial settings depends on two factors. First, existing second-order FVM frameworks may not be able to easily implement corresponding high-order methods; rewriting a mature industrial software architecture is a tedious and costly process. Second, the computational efficiency of high-order unstructured grid schemes for improving accuracy must surpass that of a simple grid-refinement strategy. In other words, they must significantly reduce computational costs while maintaining comparable accuracy. Otherwise, industrial software lacks the necessity and motivation to adopt high-order schemes. One method to satisfy these two factors is to use gradient approximations to obtain the required far-field DOFs for high-order schemes. The use of gradient estimation for far-field cell information in unstructured grids originated with Darwish and Moukalled [9], who implemented a TVD limiter. They used the gradient from neighboring cells to compute information for upstream cells. This has become the standard method for the majority of industrial CFD software based on unstructured collocated mesh frameworks to implement TVD schemes. Building on this, Sheng et al. [38] utilized a uniform-grid-based least-squares gradient-approximation method to construct a WENO scheme on unstructured grids. However, these gradient-

approximation methods present two major challenges. First, the computational formulas were derived based on uniform orthogonal Cartesian grids, and errors may occur even in marginally more complex grid types, thereby limiting the applicability of the method. This error is less pronounced in lower-order TVD, but becomes more prominent in higher-order schemes with wider templates. Second, researchers have attempted to use more accurate methods such as the least-squares approximation to improve the accuracy of gradient calculations. However, changes in the gradient values have caused the estimated cell-averaged values (FVM) of far-field cells to deviate from their true values. Therefore, these uniform-grid gradient-approximation methods perform well only when solving on uniform hexahedral/quadrilateral grids and may experience unpredictable accuracy-degradation issues when applied to other grid types, particularly boundary-layer meshes with significant local size variations [52]. Consequently, such methods have not been widely adopted by low-order FVM software to improve accuracy.

To enhance the solution accuracy of low-order FVM frameworks, this study proposes a method called data order-lifted inversion of neighbor cells (DOLINC). This method can implement various high-order spatial discretization schemes, including the fixed-template reconstruction and ENO/WENO methods, within the framework of an unstructured mesh-based finite-volume method. The fundamental concept of the proposed method is to convert the DOFs required for high-order schemes from multiple grid cells into differential DOFs stored in face-neighboring cells. During the reconstruction/interpolation of face values, the accurate inversion or retrieval of the original far-field information is achieved using strictly derived formulas. This method can be applied to any polyhedral mesh, achieving a practical accuracy no less than those of second-order high-order schemes for solving multi-dimensional flow problems. On non-uniform hexahedral meshes, particularly boundary-layer grids with significant local size variations, this method can accurately reach the theoretical convergence order of high-order schemes. The proposed DOLINC method can be seamlessly integrated into existing low-order FVM frameworks without requiring a special boundary treatment, making it adaptable to large-problem parallel computations. As an illustration, we implemented the DOLINC method based on the open-source C++ library OpenFOAM at a low level and conducted a series of computational tests for various schemes.

The remainder of this paper is organized as follows. Section 2 provides a detailed description of the core concepts and implementation principles of the DOLINC method. It introduces the scope of application of the method and its various characteristics. Section 3 demonstrates the practical performance of high-order schemes implemented based on the DOLINC method in various test cases. Section 3.1 investigates the accuracy of the scheme on a non-uniform boundary layer mesh, and Section 3.2 studies linear advection problems. Section 3.3 explores the performance of high-order methods in handling discontinuities in solving the Burgers equation. In Section 3.4, high-order schemes are applied to solve the Euler equations, encompassing four categories of one-dimensional/two-dimensional problems to comprehensively assess the computational accuracy and cost-effectiveness of the DOLINC method. Finally, Section 4 summarizes this study.

2. Description of the DOLINC method

This section provides a detailed explanation of how the DOLINC method implements high-order finite-volume schemes based on an unstructured collocated mesh solver. The applicability and computational efficiency of this method are briefly discussed.

2.1. Unstructured-Mesh-Based FVM framework

To date, many studies have demonstrated the differences in structured and unstructured grids. This study delineates unstructured mesh solvers based on the solver's behaviors in creating a grid topology and acquiring data at a low level, rather than relying on the shape of the grid elements, as illustrated in Fig. 1. When using an unstructured mesh architecture for partial differential equation (PDE) solvers to obtain grid-topology information and cell data, even a hexahedral Cartesian grid must be transformed into an unstructured storage format. In Fig. 1, the two types of grids have identical shapes (three-dimensional hexahedral grid or two-dimensional quadrilateral grid); however, different topology-storage methods determine the distinctions between the structured and unstructured grid solvers. For structured grid solvers, the global grid topology is known, enabling the solver to directly access neighboring cell information and determine the relative positions of any two cells in global coordinates. In data storage, this characteristic is typically reflected in adjacent memory-storage units corresponding to adjacent grid cells. Although the storage structure of structured grids is easy to implement for high-order schemes, it is limited by inefficiency in massive grid storage, additional considerations for process/thread topology information communications during parallel computing, and the inability to handle more complex geometric/grid forms. Therefore, primary industrial software, such as the commercial software ANSYS Fluent and STAR-CCM+, as well as open-source codes, such as OpenFOAM and code_saturne, adopt the unstructured mesh information-storage method. Unstructured meshes treat each grid cell as an independent unit, and the solver cannot directly acquire information about the neighboring cells. It relies on topological information to reconstruct relationships between cells. For example, in Fig. 1(*b*), I_o does not inherently know its neighboring cells; it needs to confirm that I_* is a neighboring cell through the connection information on the grid faces. At the data-storage level, this is reflected in adjacent storage units that do not necessarily contain information about adjacent grid cells. The DOLINC method was developed based on the unstructured grid solvers depicted in Fig. 1(*b*) and serves as an efficient approach for implementing high-order FVM schemes.

Based on the same data-storage method, unstructured meshes can be classified into several types according to the differences in their grid cell shapes, as illustrated in Fig. 2. The simplest form of an unstructured mesh is the uniform hexahedral (three-dimensional)/quadrilateral (two-dimensional) mesh shown in Fig. 2(*a*), which is geometrically identical to a classical structured mesh. The gradient-approximation methods used in previous studies [38,52] apply only to uniform unstructured grids without order or accuracy degeneration. Clearly, the applicability of this grid form is limited because the assumption of uniformity cannot be

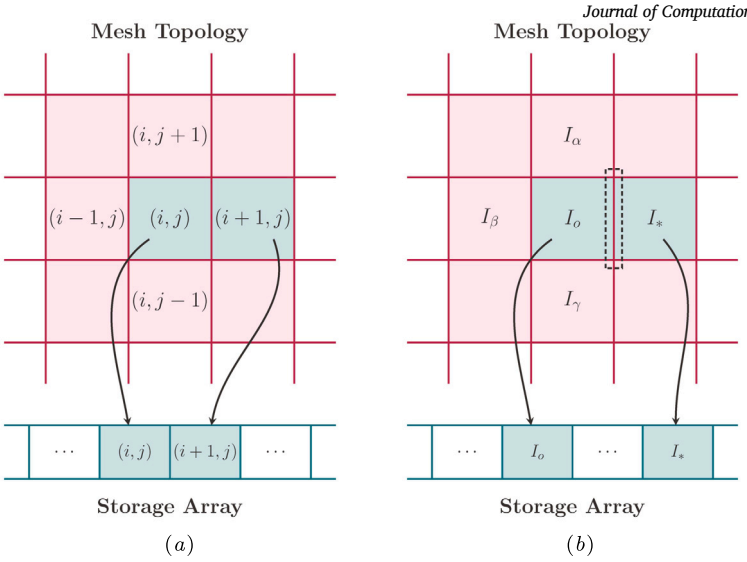

Fig. 1. Differences between structured and unstructured mesh solvers. (*a*) Information indexing based on structured mesh storage, and (*b*) information indexing based on unstructured grid storage.

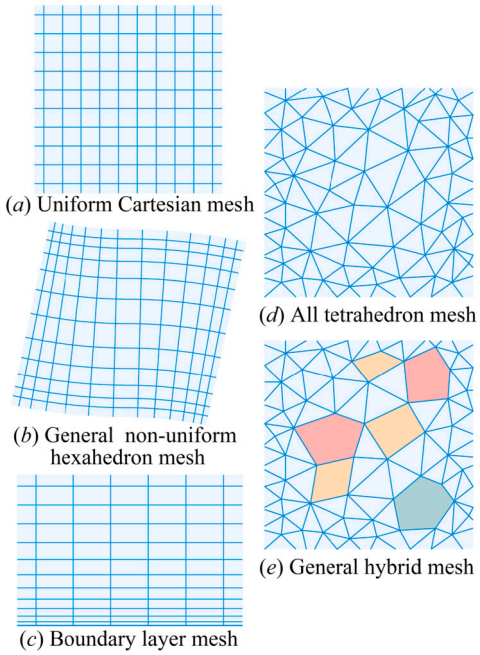

Fig. 2. Different shapes of unstructured grids. (*a*) Uniform Cartesian mesh, (*b*) non-uniform hexahedral (three-dimensional)/quadrilateral (two-dimensional) mesh, (*c*) boundary layer mesh, (*d*) pure tetrahedral (three-dimensional)/triangular (two-dimensional) mesh, (*e*) hybrid polyhedral mesh.

guaranteed in practical applications. The more common form of the non-uniform hexahedral grid used in various CFD applications is depicted in Fig. 2(*b*), exhibiting significant non-uniformity. Gradient-approximation methods neglect this non-uniformity, leading to substantial errors in high-order reconstruction, particularly in the computation of boundary-layer meshes, as shown in Fig. 2(*c*). The accurate computation of boundary-layer flows is a fundamental and crucial aspect in most CFD applications. Even in simulations involving complex geometries, such as racing cars or airplanes with polyhedral meshes, as shown in Fig. 2(*d*) or Fig. 2(*e*), boundary-layer grids are still required near the wall region to correctly resolve near-wall boundary layer flows, ensuring the reliability of the computed drag forces and other results.

Therefore, ensuring the computational accuracy of the boundary-layer meshes is a key aspect in addressing high-order schemes in unstructured-mesh-based finite volume methods. The core issue is the consideration of the local non-uniformity of the mesh. Existing gradient-approximation methods do not consider the size variation of the mesh, leading to uncontrollable accuracy degradation in unstructured meshes other than that depicted in Fig. 2(*a*) [9,38]. To guarantee the theoretical accuracy of high-order methods on non-

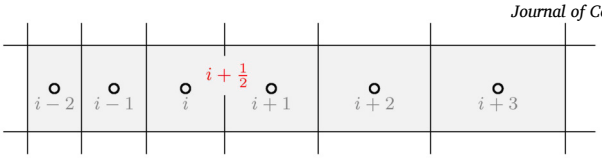

Fig. 3. Schematic of non-uniform split multi-dimensional mesh (three-dimensional hexahedral/two-dimensional quadrilateral).

uniform meshes, the proposed DOLINC method employs a strictly mathematically derived result that considers the local mesh growth rate when retrieving far-field cell information. This approach provides accurate results for meshes resembling those in Figs. 2(*a*) and 2(*b*). Similar to the gradient methods, DOLINC can implement high-order schemes on all types of unstructured meshes, as depicted in Fig. 2, that is, it is applicable to any polyhedral mesh.

2.2. Spatial discretization of finite-volume method

The derivation of the mathematical model of the DOLINC method is based on the spatial discretization of the finite-volume method, as shown in Fig. 3, where the mesh size exhibits significant local variations. Fig. 3 shows a "split" or "x-split" three-dimensional analysis on which the following derivations are based; that is, physical quantities and spatial coordinates are all three-dimensional while the face reconstruction procedure can be handled in a quasi-one-dimensional manner. This split multi-dimensional approach was used for most derivations in the monograph of Toro [44]. For the unstructured grids with cells of non-uniform hexahedral shapes, which are the focus of this study, the results obtained employing this analysis are exact, without approximation errors. For mesh skewness or non-orthogonality that could disrupt this quasi-one-dimensional behavior, the correction strategies commonly used in conventional FVM methods could be adopted. In the figure, $i+1/2$ marks the current face requiring interpolation/reconstruction, whereas i marks the neighboring mesh cell corresponding to its upwind direction. Assume that the growth rate of the local grid cells remains constant; thus,

$$g = \frac{\Delta x_{j+1}}{\Delta x_j} = \text{const.}, \quad j = \ldots, i-1, i, i+1, \ldots \tag{1}$$

where g represents the cell growth rate, and its value is in the range of $(0, \infty)$. For three-dimensional grids, the center-to-center distance between two adjacent cells on a mesh face is

$$\begin{aligned} &\mathbf{d} = \mathbf{x}_{i+1} - \mathbf{x}_i, \\ &d = \|\mathbf{d}\|. \end{aligned} \tag{2}$$

In the low-order FVM framework of industrial software, the incorporation of spatial discretization schemes involves the use of weights ω to introduce the influence of distance variation between the cell center and mesh-face center:

$$\omega = \frac{||\mathbf{x}_{i+1} - \mathbf{x}_{i+\frac{1}{2}}||}{||\mathbf{x}_i - \mathbf{x}_{i+\frac{1}{2}}|| + ||\mathbf{x}_{i+1} - \mathbf{x}_{i+\frac{1}{2}}||}. \tag{3}$$

The above equation shows that the relationship between the grid growth rate g and ω can be obtained as follows:

$$g = \frac{\omega}{1-\omega} = \frac{||\mathbf{x}_{i+1} - \mathbf{x}_{i+\frac{1}{2}}||}{||\mathbf{x}_{i+\frac{1}{2}} - \mathbf{x}_i||}. \tag{4}$$

Assuming the exact solution of the PDE in the studied region is the function $\phi(x)$ defined on the domain, the FVM solver obtains the average value $\overline{\phi}_i$ within the grid cell. For the x-split three-dimensional problem, we have

$$\overline{\phi}_i = \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} \phi(\xi)\, \mathrm{d}\xi. \tag{5}$$

In the derivation of the reconstruction formula, the primitive function is used:

$$\Phi(x) = \int_{x_0}^{x} \phi(\xi)\, \mathrm{d}\xi, \tag{6}$$

where x_0 is an arbitrarily selected reference point. After obtaining the interpolation polynomial $P(x)$ of $\Phi(x)$, the derivative polynomial $p(x)$ is obtained. It can be proved that (see literature of Shu [40] for more details)

$$\overline{\phi}_i = \int_{x_{i-\frac{1}{2}}}^{x_{i+\frac{1}{2}}} p(\xi)\,\mathrm{d}\xi. \tag{7}$$

Therefore, the interpolation polynomial $p(x)$ can be used to construct function values or various order derivatives on the mesh faces.

2.3. Basic principles of DOLINC method for inverting far-field data

Based on the previous discussion, the main challenge in achieving universal high-order schemes on unstructured grids is efficiently obtaining relevant data from far-field elements or their corresponding multi-DOF information. The computational cost of high-order reconstruction methods is undoubtedly higher than that of low-order FVM methods because of the necessity of additional DOFs when constructing interpolation polynomials. Therefore, a feasible high-order unstructured-grid finite-volume method must meet three efficiency requirements: a) easy implementation based on the current low-order FVM; b) minimal additional runtime storage; and c) a short time needed to acquire additional DOFs.

The fundamental idea behind the DOLINC method is to store DOF-transformed order-lifted differentiations and precisely invert the original data based on this storage. This corresponds to two core steps. The first step involves storing multi-grid information from the high-order reconstruction template as localized high-order differential data, and the second step utilizes the stored differential data to accurately calculate and recover the transformed DOFs. Throughout the computation process, the grid topology or solver matrix assembly within the low-order FVM framework does not require modification. All computational steps are completed within the interpolation/reconstruction phase of the equation discretization.

In the second-order FVM, linear interpolation combined with Gauss's theorem is commonly employed to calculate the gradient of grid cells (referred to as the central differencing gradient or central differencing derivative). Considering the local non-uniform variation of the mesh cell, the following exact expression can be derived:

$$\overline{\phi}_{i+n+2} = g(\overline{\phi}_{i+n} + 2g^n d_j\,\overline{\phi}_{i+n+1,j}) + (1-g)\overline{\phi}_{i+n+1}. \tag{8}$$

Here, $n = 0, 1, 2, \ldots$, and d_j is the distance vector of adjacent cell centers i and $i+1$ calculated using Equation (2). To ensure more concise notation in the derivation results, the tensor notation convention recommended by Aris [4] was adopted for derivations of the DOLINC approach. Spatial partial derivatives of tensors employing this convention are represented with symbols following a comma in the subscript to save redundant notations. $\overline{\phi}_{i+n+1,j}$ represents the j-th component of the first-order central differencing derivative of $\overline{\phi}_{i+n+1}$, i.e.,

$$\overline{\phi}_{i+n+1,j} = \frac{\partial \overline{\phi}_{i+n+1}}{\partial x_j}. \tag{9}$$

Further details for obtaining Equation (8) can be found in Appendix A. In Equation (8), the right-hand side only includes information from grid cells $i+n$ and $i+n+1$, successfully obtaining the average value of the far-field cells on the left-hand side of the equation using only first-order derivatives. Note that the central differencing gradient in Equation (8) appears in pairs with the vector of the cell-center distances and can be expressed in a differential form. We define a representation called the DOLINC differential as follows:

$$\begin{aligned} \varphi_n^{(0)} &= \varphi_n = \overline{\phi}_{i+n}, \\ \varphi_n^{(m)} &= 2^m \overline{\phi}_{i+n,j_1 j_2 \cdots j_m} d_{j_1} d_{j_2} \cdots d_{j_m}, \quad m = 1, 2, 3, \ldots \end{aligned} \tag{10}$$

Here, $\overline{\phi}_{i+n,j_1 j_2 \cdots j_m}$ represents the m-order partial derivative of $\overline{\phi}_{i+n}$, an m-order higher tensor:

$$\overline{\phi}_{i+n,j_1 j_2 \cdots j_m} = \frac{\partial^m \overline{\phi}_{i+n}}{\partial x_{j_m} \cdots \partial x_{j_2} \partial x_{j_1}}. \tag{11}$$

Note that an m-time tensor contraction with distance vector $\mathbf{d}$ occurs in Equation (10), that is,

$$\overline{\phi}_{i+n,j_1 j_2 \cdots j_m} d_{j_1} d_{j_2} \cdots d_{j_3} = \left(\left(\left(\frac{\partial^m \overline{\phi}_{i+n}}{\partial x_{j_m} \cdots \partial x_{j_2} \partial x_{j_1}} d_{j_1} \right) d_{j_2} \right) \cdots d_{j_m} \right). \tag{12}$$

Equation (8) can be further extended to include a recursive relationship involving high-order differential information, as follows:

$$\varphi_{n+2}^{(m)} = g(\varphi_n^{(m)} + g^n \varphi_{n+1}^{(m+1)}) + (1-g)\varphi_{n+1}^{(m)}. \tag{13}$$

A detailed derivation of the corresponding differentials and the reason why this symbol convention was empolyed can be found in Appendix A as well. Similarly, for the mesh on the side of $n = 0, -1, -2, \ldots$, we obtain

$$\varphi_{n-1}^{(m)} = g^{-1}(\varphi_{n+1}^{(m)} - g^n \varphi_n^{(m+1)}) + (1-g^{-1})\varphi_n^{(m)}. \tag{14}$$

These two recursive expressions can ultimately be unified into a universal DOLINC differentiation relationship:

$$\varphi_{n+1}^{(m)} - g\varphi_{n-1}^{(m)} = g^n \varphi_n^{(m+1)} + (1-g)\varphi_n^{(m)}, \quad m = \ldots, -2, -1, 0, 1, 2, 3, \ldots \tag{15}$$

Through multiple recursive applications of Equation (15), any far-field cell information can be transformed into the differential information of various-order $\varphi_0^{(m)}$ and $\varphi_1^{(m)}$ on the mesh face of two adjacent grid cells. This transformation can be used for constructing high-order schemes. In terms of DOFs, the originally required 0th-order wide-range DOFs are transformed into local high-order ones, and the number of DOFs remains unchanged. In terms of accuracy, by neglecting floating-point errors in the calculation, the calculated $\varphi_{n-1}^{(m)}$ and $\varphi_{n+1}^{(m)}$ on the left-hand side of Equation (15) are consistent with the original values $\varphi_{n-1}^{(m)}$ and $\varphi_{n+1}^{(m)}$ stored on the far-field mesh cells involved in the computations of $\varphi_n^{(m+1)}$ on the right-hand side. Notably, the significance of DOLINC differentials is in providing a medium or an intermediate variable for retrieving far-field cell information, and they do not require the approximation of the exact values of the actual function differentials. In contrast, using more accurate gradient-calculation methods, such as the least-squares approximation, introduces errors to the right-hand side of Equation (15), leading to a deviation of the recalculated far-field data from the initial values. Additionally, using the exact average values of the function gradient in the calculation formula still results in a decrease in the accuracy of high-order schemes. This is the reason we refer to this process as an inversion rather than a gradient approximation: the accuracy of the result depends on whether the inversion formula is strictly valid, not on how accurate the estimated gradient used is. In terms of storage, DOLINC differentials and the physical quantity ϕ to be solved belong to the same-order tensors, requiring the same data length, whereas the tensor order increases continuously when calculating gradients of ϕ. For a D-dimensional problem, the data storage required for calculating the Nth-order DOLINC differentials is N+1 times that of the original, and the data-storage required for calculating Nth-order gradients is $\frac{D^{N+1}-1}{D-1}$ times that of the original.

It should be noted that arbitrary far-field cells were converted to high-order differentials stored in two face-adjacent cells, enabling the construction of schemes of an arbitrary high order. The only difference is that higher-order schemes require high-order DOLINC differentials to be computed and stored in advance. Without loss of generality, schemes using at most second-order differentials are presented for demonstration of the DOLINC method in this study. When the required data for the corresponding high-order scheme do not exceed the split four-grid-cell template, only the storage of first-order DOLINC differentials is necessary. The inversion formula is as follows:

$$\varphi_{-1} = g^{-1}(\varphi_1 - \varphi_0^{(1)}) + (1-g^{-1})\varphi_0 = g^{-1}\varphi_1^{(0)} + (1-g^{-1})\varphi_0^{(0)} - g^{-1}\varphi_0^{(1)}, \tag{16}$$

$$\varphi_2 = g(\varphi_0 + \varphi_1^{(1)}) + (1-g)\varphi_1 = g\varphi_0^{(0)} + (1-g)\varphi_1^{(0)} + g\varphi_1^{(1)}. \tag{17}$$

At this point, the split-mesh template $\{\varphi_{-1}, \varphi_0, \varphi_1, \varphi_2\}$ is transformed into a DOLINC template $\{\varphi_0^{(0)}, \varphi_0^{(1)}, \varphi_1^{(0)}, \varphi_1^{(1)}\}$ with the same number of DOFs. This type of high-order scheme, which only requires the use of first-order DOLINC differentials, is referred to as a first-order DOLINC scheme (the order of the DOLINC scheme is different from the convergence order of the high-order scheme; for example, a fourth-order accurate fixed-template reconstruction scheme belongs to the first-order DOLINC scheme). To achieve a form similar to conventional unstructured-mesh finite-volume schemes, the inversion formula for the far-field cells in the DOLINC scheme is organized into the following general form:

$$\overline{\phi}_j = a_0^{j-i}\overline{\phi}_{(i)} + a_1^{j-i}\overline{\phi}_{(i+1)} + A_\delta^{j-i}, \quad j = \ldots, i-2, i-1, i+2, i+3, \ldots \tag{18}$$

Thus, for first-order DOLINC schemes, we have

$$\begin{aligned} a_0^{-1} &= 1 - g^{-1}, \\ a_1^{-1} &= 1 - a_0^{-1}, \\ A_\delta^{-1} &= -g^{-1}\varphi_0^{(1)}, \end{aligned} \tag{19}$$

and

$$\begin{aligned} a_0^2 &= g, \\ a_1^2 &= 1 - a_0^2, \\ A_\delta^2 &= g\varphi_1^{(1)}. \end{aligned} \tag{20}$$

When high-order reconstruction requires the use of second-order DOLINC differentials (second-order DOLINC scheme), such as in the case of a fixed central-template reconstruction with sixth-order accuracy, the additional inversion formula for the far-field cells is as follows:

$$\varphi_{-2} = g^{-1}\varphi_0^{(0)} + (1-g^{-1})\varphi_{-1}^{(0)} - g^{-2}\varphi_{-1}^{(1)}, \tag{21}$$

$$\varphi_3 = g\varphi_1^{(0)} + (1-g)\varphi_2^{(0)} + g^2\varphi_2^{(1)}, \tag{22}$$

where

$$\varphi_{-1}^{(0)} = g^{-1}\varphi_1^{(0)} + (1-g^{-1})\varphi_0^{(0)} - g^{-1}\varphi_0^{(1)}, \tag{23}$$

$$\varphi_{-1}^{(1)} = g^{-1}\varphi_1^{(1)} + (1-g^{-1})\varphi_0^{(1)} - g^{-1}\varphi_0^{(2)}, \tag{24}$$

$$\varphi_2^{(0)} = g\varphi_0^{(0)} + (1-g)\varphi_1^{(0)} + g\varphi_1^{(1)}, \tag{25}$$

$$\varphi_2^{(1)} = g\varphi_0^{(1)} + (1-g)\varphi_1^{(1)} + g\varphi_1^{(2)}. \tag{26}$$

Substituting Equations (23)–(26) into Equations (21) and (22), the inversion formulas become

$$\begin{aligned}\varphi_{-2} = {} & g^{-1}(1-g^{-1})\varphi_1^{(0)} - g^{-3}\varphi_1^{(1)} \\ & + (1-g^{-1}+g^{-2})\varphi_0^{(0)} - g^{-1}(1-g^{-2})\varphi_0^{(1)} + g^{-3}\varphi_0^{(2)},\end{aligned} \tag{27}$$

and

$$\begin{aligned}\varphi_3 = {} & g(1-g)\varphi_0^{(0)} + g^3\varphi_0^{(1)} \\ & + (1-g+g^2)\varphi_1^{(0)} + g(1-g^2)\varphi_1^{(1)} + g^3\varphi_1^{(2)}.\end{aligned} \tag{28}$$

At this point, the split six-grid template $\{\varphi_{-2}, \varphi_{-1}, \varphi_0, \varphi_1, \varphi_2, \varphi_3\}$ is transformed into a DOLINC template $\{\varphi_0^{(0)}, \varphi_0^{(1)}, \varphi_0^{(2)}, \varphi_1^{(0)}, \varphi_1^{(1)}, \varphi_1^{(2)}\}$ with the same number of DOFs. By introducing the general form of Equation (18), we obtain

$$\begin{aligned} a_0^{-2} &= (1-g^{-1}+g^{-2}), \\ a_1^{-2} &= 1 - a_0^{-2}, \\ A_\delta^{-2} &= -g^{-3}\varphi_1^{(1)} - g^{-1}(1-g^{-2})\varphi_0^{(1)} + g^{-3}\varphi_0^{(2)}, \end{aligned} \tag{29}$$

and

$$\begin{aligned} a_0^{3} &= g - g^2, \\ a_1^{3} &= 1 - a_0^{3}, \\ A_\delta^{3} &= g^3\varphi_0^{(1)} + g(1-g^2)\varphi_1^{(1)} + g^3\varphi_1^{(2)}. \end{aligned} \tag{30}$$

Following the inversion procedure for far-field elements using the aforementioned DOLINC method, the corresponding DOLINC version of high-order schemes can be constructed based on expressions for high-order reconstruction. It should be noted that although only detailed derivation of DOLINC differentials and inversion formulas of φ_{-1}, φ_2, φ_{-2} and φ_3 are presented in this section, one can similarly obtain inversion expressions of φ_{-3}, φ_4, φ_{-4}, $\varphi_5, \cdots$ using Equations (13), (14), and (15) in a recursive manner. Therefore, implementing schemes of even higher order would only require additional mathematical and coding work without any methodological constraints.

2.4. Implementation of high-order schemes based on DOLINC method (DOLINC schemes)

Generally, if the index of the leftmost cell adopted by a reconstruction template is denoted as $i-r$, then the kth-order reconstruction method can be represented as

$$\phi_{i+\frac{1}{2}} = p(x_{i+\frac{1}{2}}) = \sum_{j=0}^{k-1} c_{rj}^k \overline{\phi}_{i-r+j}, \tag{31}$$

where

$$c_{rj}^k = \left[\sum_{m=j+1}^{k} \frac{\sum_{\substack{l=0 \\ l\neq m}}^{k} \prod_{\substack{q=0 \\ q\neq m,l}}^{k} \left(x_{i+\frac{1}{2}} - x_{i-r+q-\frac{1}{2}} \right)}{\prod_{\substack{l=0 \\ l\neq m}}^{k} \left(x_{i-r+m-\frac{1}{2}} - x_{i-r+l-\frac{1}{2}} \right)} \right] \Delta x_{i-r+j}. \tag{32}$$

Detailed derivations are available in the literature [40]. For simplicity, the superscripts can be omitted when the order of reconstruction is clear, and the coefficient is represented by c_{rj}.

Using the third-order reconstruction of $k=3, r=1$ as an example, a corresponding DOLINC scheme was constructed. In this case, considering the non-uniformity of the grid, we can derive through Equation (32) that

$$c_{10} = \frac{-g^3}{G_2 G_3}, \tag{33}$$

$$c_{11} = \frac{g(1+2g+2g^2)}{G_2 G_3}, \tag{34}$$

$$c_{12} = \frac{1}{G_3}. \tag{35}$$

In the above expression, G_n is employed to simplify the coefficient representation, and its specific definition is given by

$$\begin{aligned} G_n &= 1 + g + g^2 + g^3 + \cdots + g^{n-1} \\ &= \frac{g^n - 1}{g - 1}, \quad n = 1, 2, 3, \ldots \end{aligned} \tag{36}$$

Noticing that when $g = 1$, i.e., the grid size remains unchanged, $G_n = n$, the coefficients degenerate to the constants -1/6, 5/6, and 1/3. This non-uniform mesh third-order reconstruction employs a fixed template one-cell biased towards the upwind direction. Hereafter, we denote this reconstruction scheme as the third-order upwind stencil reconstruction (USR3). The basic representation of USR3 can be organized into the weight-representation method commonly used in lower-order FVM using Equation (19):

$$\begin{aligned} \phi_{i+\frac{1}{2}} &= c_{10}\overline{\phi}_{i-1} + c_{11}\overline{\phi}_i + c_{12}\overline{\phi}_{i+1} \\ &= (c_{10}a_0^{-1} + c_{11})\overline{\phi}_i + (c_{10}a_1^{-1} + c_{12})\overline{\phi}_{i+1} + c_{10}A_\delta^{-1} \\ &= w_s\overline{\phi}_i + (1 - w_s)\overline{\phi}_{i+1} + c_r, \end{aligned} \tag{37}$$

where

$$w_s = c_{10}a_0^{-1} + c_{11}, \tag{38}$$

$$c_r = c_{10}A_\delta^{-1}. \tag{39}$$

This representation style, derived for high-order schemes through the DOLINC approach, is referred to as the DOLINC representation. In Equation (37), the DOLINC representation of USR3 is consistent with the logical form commonly used in lower-order FVM codes for face-interpolation schemes, requiring no additional adjustments to its architecture. In practical implementation, the USR3 scheme can be realized through an embedding approach similar to the second-order upwind or TVD schemes. If time advancement is explicitly conducted in solving the PDEs [20], the DOLINC-implemented USR3 is entirely equivalent to the classical third-order accurate reconstruction. When employing an implicit time-integration scheme for PDEs, as deduced above, the DOLINC method essentially provides a modified semi-implicit third-order scheme that contributes to the stability of the matrix to be solved.

Owing to the lack of advanced high-order schemes, industrial software typically recommends the use of a second-order central differencing scheme (CD2) to mitigate the impact of numerical dissipation on the results of turbulence LESs or DNSs [14]. Therefore, we consider the example of a fourth-order central stencil reconstruction (CSR4) scheme to provide the corresponding DOLINC representation; it shares similarities with CD2, but offers higher accuracy. In this case, $k = 4, r = 1$, and through derivation, we obtain

$$\begin{aligned} \phi_{i+\frac{1}{2}} &= c_{10}\overline{\phi}_{i-1} + c_{11}\overline{\phi}_i + c_{12}\overline{\phi}_{i+1} + c_{13}\overline{\phi}_{i+2} \\ &= w_s\overline{\phi}_i + (1 - w_s)\overline{\phi}_{i+1} + c_r, \end{aligned} \tag{40}$$

where

$$w_s = c_{10}a_0^{-1} + c_{11} + c_{13}a_0^2, \tag{41}$$

$$c_r = c_{10}A_\delta^{-1} + c_{13}A_\delta^2. \tag{42}$$

In the above equation, the basic representation of CSR4 also considers the non-uniformity of the mesh for each reconstruction coefficient, and the calculation formula is as follows:

$$c_{10} = \frac{-g^5}{G_3G_4}, \tag{43}$$

$$c_{11} = \frac{g^2(1 + 2g + 2g^2 + 2g^3)}{G_3G_4}, \tag{44}$$

$$c_{12} = \frac{2 + 2g + 2g^2 + g^3}{G_3G_4}, \tag{45}$$

$$c_{13} = \frac{-1}{G_3G_4}. \tag{46}$$

Similarly, when $g = 1$, the above coefficients degenerate to the constants -1/12, 7/12, 7/12, and -1/12.

USR3 and CSR4 in the above derivation process only involve the DOLINC inversion coefficients from Equations (19) and (20), making them first-order DOLINC schemes. Second-order DOLINC schemes such as USR5 or CSR6 would require the use of DOLINC inversion coefficients from Equations (29) and (30). Based on the DOLINC fixed-stencil schemes, achieving DOLINC versions of the ENO and WENO methods only requires supplementing stencil-smoothness indicators or a stencil-weight-calculation approach. Similarly, the calculation of indicators or weights must consider local variations in grid sizes. For example, in the third-order accurate

ENO method (ENO3) using the classical stencil-selection approach [23], the smoothness indicator for the required split two-grid stencil is

$$\begin{aligned}\Delta_0^2\boldsymbol{\Phi}[x_{i-\frac{3}{2}},x_{i-\frac{1}{2}},x_{i+\frac{1}{2}}] &= g(\overline{\phi}_i-\overline{\phi}_{i-1})\\ &= \overline{\phi}_i-\overline{\phi}_{i+1}+\varphi_0^{(1)},\end{aligned} \tag{47}$$

$$\Delta_0^2\boldsymbol{\Phi}[x_{i-\frac{1}{2}},x_{i+\frac{1}{2}},x_{i+\frac{3}{2}}] = \overline{\phi}_{i+1}-\overline{\phi}_i. \tag{48}$$

Similarly, the smoothness indicator for the split three-grid stencil is obtained as follows:

$$\Delta_{-1}^3\Delta_0^2\boldsymbol{\Phi}[x_{i-\frac{5}{2}},x_{i-\frac{3}{2}},x_{i-\frac{1}{2}},x_{i+\frac{1}{2}}] = g^2[\overline{\phi}_i-(1+g)\overline{\phi}_{i-1}+g\overline{\phi}_{i-2}], \tag{49}$$

$$\Delta_{-1}^3\Delta_0^2\boldsymbol{\Phi}[x_{i-\frac{3}{2}},x_{i-\frac{1}{2}},x_{i+\frac{1}{2}},x_{i+\frac{3}{2}}] = \overline{\phi}_{i+1}-(1+g)\overline{\phi}_i+g\overline{\phi}_{i-1}, \tag{50}$$

$$\Delta_{-1}^3\Delta_0^2\boldsymbol{\Phi}[x_{i-\frac{1}{2}},x_{i+\frac{1}{2}},x_{i+\frac{3}{2}},x_{i+\frac{5}{2}}] = g^{-2}[\overline{\phi}_{i+2}-(1+g)\overline{\phi}_{i+1}+g\overline{\phi}_i]. \tag{51}$$

The indicator calculation formulas above require the use of the corresponding DOLINC inversion formulas. Similarly, by incorporating stencil-weight calculations, WENO reconstruction methods of various orders of accuracy can be obtained. ENO3-DOLINC and WENO5-DOLINC (hereafter distinguished by this representation to denote the same high-order scheme implemented using different methods) involve multiple alternative stencils; thus, they are classified as second-order DOLINC schemes.

2.5. Error analysis of uniform-mesh-based least-squares gradient-approximation method

Based on the previous derivation, the DOLINC scheme can achieve a theoretical convergence accuracy for the corresponding reconstruction. Two key points are notable in this regard: first, both the inversion formula and scheme coefficients in the method must consider local mesh growth, and second, the DOLINC differentials must be rigorously calculated. Owing to its inability to ensure these two points, the uniform-mesh least-squares gradient-approximation method (hereinafter referred to as the UG method) may exhibit uncontrollable accuracy degradation in the finite-volume scheme obtained. It should be noted that the term "uniform-mesh-based" refers to the assumption underlying the inversion formulas in the UG method—how the gradient data are used to compute values at distant cells, rather than the assumption underlying the gradient computation itself. In the UG Method, the least-squares approximation used to compute gradients is implemented for non-uniform, unstructured meshes rather than uniform ones, often resulting in more accurate gradient prediction than the linear interpolation employed by the DOLINC approach.

The next step involves an error analysis of the UG method through mathematical derivation. First, we consider the influence of non-uniformity. Let the reconstructed values obtained using the DOLINC method be denoted as $\phi_{i+\frac{1}{2}}^{DLC}$. For a kth-order reconstruction, we have

$$\phi_{i+\frac{1}{2}}^{DLC} = \sum_{j=0}^{k-1} c_{rj}^{DLC}\,\overline{\phi}_{i-r+j}. \tag{52}$$

For the non-uniform grid exhibited in Fig. 3, we have

$$\phi_{i+\frac{1}{2}} = \phi_{i+\frac{1}{2}}^{DLC} + O(\Delta x^k), \tag{53}$$

where $\phi_{i+\frac{1}{2}}$ represents the exact function values. Similarly, if we follow the UG method and assume a uniform mesh to calculate various coefficients, a reconstruction method can be obtained (temporarily assuming no errors in the estimation of the phantom points in the UG method). The reconstruction results obtained using this method are denoted as $\phi_{i+\frac{1}{2}}^{UG}$. At this point, the UG scheme can also be expressed as

$$\phi_{i+\frac{1}{2}}^{UG} = \sum_{j=0}^{k-1} c_{rj}^{UG}\,\overline{\phi}_{i-r+j}. \tag{54}$$

This non-DOLINC method induces the error

$$\phi_{i+\frac{1}{2}}^{UG} - \phi_{i+\frac{1}{2}}^{DLC} = \sum_{j=0}^{k-1}\left(c_{rj}^{UG}-c_{rj}^{DLC}\right)\overline{\phi}_{i-r+j}. \tag{55}$$

Let $\overline{\phi}_{i-r+j} = \phi_0 + \delta\phi_{i-r+j}$, where ϕ_0 is an arbitrary reference value that can be considered the exact value $\phi_{i+\frac{1}{2}}$. Notice that

$$\sum_{j=0}^{k-1}\left(c_{rj}^{UG}-c_{rj}^{DLC}\right) = 0, \tag{56}$$

then, we have

$$\begin{aligned}\phi^{UG}_{i+\frac{1}{2}} - \phi^{DLC}_{i+\frac{1}{2}} &= \sum_{j=0}^{k-1}\left(c^{UG}_{rj} - c^{DLC}_{rj}\right)\left(\phi_0 + \delta\phi_{i-r+j}\right),\\ &= \sum_{j=0}^{k-1}\left(c^{UG}_{rj} - c^{DLC}_{rj}\right)\delta\phi_{i-r+j}.\end{aligned} \tag{57}$$

Further examination of the above errors shows that the first component of the error is

$$\epsilon_c = c^{UG}_{rj} - c^{DLC}_{rj} = f(g) \sim O(1), \tag{58}$$

which is solely a function of the mesh growth rate g. Therefore, even as the grid size decreases continuously, ϵ_c will remain constant as long as non-uniformity exists in the grid. This implies that the magnitude of ϵ_c in the UG method is entirely determined by the growth rate. When the UG method is applied to a uniform grid, this error term disappears; however, when local grid sizes significantly vary, ϵ_c cannot be reduced in another manner. For the second component of the error,

$$\epsilon_\phi = \delta\phi_{i-r+j} \sim O(\Delta x), \tag{59}$$

is why mesh refinement can improve non-DOLINC schemes in practical applications. As the mesh gradually refines, the physical range of the local template decreases, leading to a reduction in the local variation of the solution within the template. However, as indicated by Equation (59), the convergence accuracy of this error term is only first-order. Introducing a first-order error term directly into a kth-order scheme results in uncontrollable accuracy degradation of the reconstruction [52].

In the above analysis, Equation (54) directly utilizes $\overline{\phi}_{i-r+j}$; however, in the UG method, $\overline{\phi}_{i-r+j}$ cannot be correctly inverted because of the fundamentally incorrect inversion caused by the use of non-DOLINC differential calculations. By employing the representation of the DOLINC scheme in Equation (40), an expression equivalent to Equation (52) can be obtained:

$$\phi^{DLC}_{i+\frac{1}{2}} = w^{DLC}_s\overline{\phi}_i + (1 - w^{DLC}_s)\overline{\phi}_{i+1} + c^{DLC}_r, \tag{60}$$

Similarly, the reconstructed value through the UG method can be transformed into

$$\phi^{UG}_{i+\frac{1}{2}} = w^{UG}_s\overline{\phi}_i + (1 - w^{UG}_s)\overline{\phi}_{i+1} + c^{UG}_r. \tag{61}$$

Then, we have

$$\phi^{UG}_{i+\frac{1}{2}} - \phi^{DLC}_{i+\frac{1}{2}} = \left(w^{UG}_s - w^{DLC}_s\right)\left(\overline{\phi}_i - \overline{\phi}_{i+1}\right) + \left(c^{UG}_r - c^{DLC}_r\right). \tag{62}$$

Similarly,

$$\epsilon_w = w^{UG}_s - w^{DLC}_s = f(g) \sim O(1), \tag{63}$$

Apart from the non-uniform errors in the reconstruction coefficients, as described in Equation (58), an additional contribution of ϵ_w exists from neglecting the non-uniformity in the calculation of phantom points in the UG method. Similarly, at this point,

$$\epsilon_\phi = \overline{\phi}_i - \overline{\phi}_{i+1} \sim O(\Delta x). \tag{64}$$

However, unlike the previous derivation, an additional term is introduced into the error at this point:

$$\epsilon_\varphi = c^{UG}_r - c^{DLC}_r = f(g, \varphi^{(1)}_0, \varphi^{(2)}_0, \cdots, \varphi^{(1)}_1, \varphi^{(2)}_1, \cdots). \tag{65}$$

From the above equation, regardless of the accuracy of the gradient calculation in c^{UG}_r, introducing a new error term is unavoidable as long as it does not adopt DOLINC differentials. When the UG method employs the least-squares approximation to compute the gradient, it may seem to obtain more accurate gradient data on the surface. However, in reality, this leads to a larger discrepancy between the calculated phantom point data and original field data, causing further uncontrollable accuracy degradation in the reconstruction scheme. Therefore, we emphasize once again that, as long as all information of the far-field element can be accurately inverted, the accuracy of the derivatives/gradients/differentials involved in the inversion formula is irrelevant. This is the core idea that distinguishes the DOLINC method from general gradient-approximation methods.

2.6. Boundary handling of DOLINC method

A good algorithm should be efficient, accurate, and stable, and the boundary handling required for the DOLINC scheme should exhibit these characteristics. For a low-order FVM framework to implement DOLINC schemes, no special treatment of the boundary mesh cells or modifications are required for the existing boundary-condition codes. The DOLINC scheme can be applied directly without any adjustments.

To handle coupled boundaries, such as processor boundaries or periodic boundaries, the key to implementation lies in ensuring accurate and efficient data exchange between the two sides of the interface. In the high-order schemes implemented by the DOLINC method, coupled boundaries still only require the communication of data between adjacent grids on both sides of the boundary,

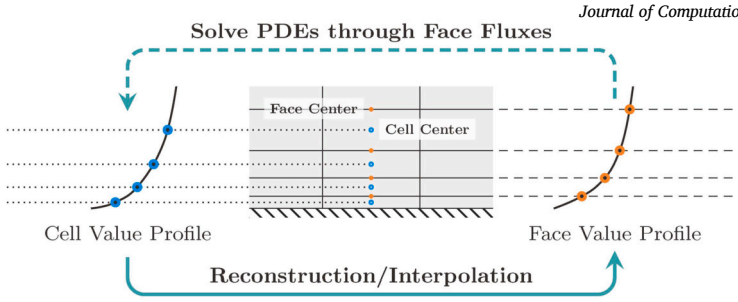

Fig. 4. Typical boundary layer mesh reconstruction/interpolation and PDE-solution iterations.

without considering far-field elements. Compared to the low-order finite-volume scheme, the only difference in the DOLINC scheme is the addition of DOLINC differentials to the exchanged data.

For non-coupled boundaries, such as the Dirichlet or Neumann boundaries, the handling approach of the low-order FVM still ensures the correct implementation of the DOLINC scheme at the boundary. In this case, the DOLINC differentials use the boundary values directly, and the inversion calculation yields auxiliary cells located on the other side of the boundary. The relationship between the auxiliary and boundary-adjacent cells strictly adheres to the boundary conditions. For example, for a given outward normal gradient q on a Neumann boundary, assuming the distance from the body center of the boundary cell to the center of the boundary face is d_{fC}, and the value of the boundary cell is T_C, the interpolated data on the boundary face is obtained as $T_f = T_C + q\, d_{fC}$. With known values of T_f and T_C, the inversion formula yields the value of the auxiliary element on the other side as $T_O = T_C + 2(T_f - T_C)$. At this point, the outward normal gradient on the boundary can be recalculated using T_O and T_C as follows:

$$\frac{\partial T}{\partial n_f} = \frac{T_O - T_C}{2d_{fC}} = \frac{T_f - T_C}{d_{fC}} = q. \tag{66}$$

Therefore, in the DOLINC method, the Neumann boundary remains unchanged. Similarly, at the Dirichlet boundary, T_f obtained through the linear interpolation of T_O and T_C remains constant. Overall, the DOLINC method does not require the treatment of non-coupled boundaries to ensure its computational accuracy and efficiency.

3. Numerical results and discussion

This section demonstrates the performance of the DOLINC method when applied to numerical solutions of different problems. Using the open-source C++ library, OpenFOAM, we successfully integrated the DOLINC method into a low-order FVM framework. We compared the computational results of the DOLINC scheme with those of classical finite-volume schemes such as second-order central differencing, second-order upwind, and TVD limiters. The various solvers used in this section employed either a fourth-order classical Runge–Kutta scheme or third-order TVD Runge–Kutta scheme to minimize the impact of temporal discretization on the final results. Details of the implementation are available in the literature [20]. The HPC clusters used for the validation cases consist of multiple computational nodes. Each node comprised two Intel® Xeon® Platinum 9242 processors, totaling 96 physical cores. The maximum memory capacity of each node was 384 GB. Additional HPC settings for the numerical examples are available in the literature [19].

3.1. Boundary layer flows on non-uniform grids

As mentioned in Section 2.5, the DOLINC reconstruction scheme exhibits higher accuracy than non-DOLINC reconstruction schemes, especially in non-uniform grids with local growth. To verify this, we considered the reconstruction process in the context of a typical boundary-layer-flow problem on a non-uniform mesh. Fig. 4 illustrates the solution iterations on a typical boundary-layer mesh. In the reconstruction phase, the solver calculated face values based on the cell averages obtained from the previous iteration or time step. The reconstructed face values in the form of the scheme influenced the update of the cell values at the current iteration or time step in the PDE-solving algorithm. The updated cell values were then used for reconstruction in the next iteration or timestep.

Two points must be noted regarding this process. First, the errors introduced by the reconstruction scheme are repeatedly incorporated into the solution process and do not gradually disappear during the computation process. In fact, when solving problems that are sensitive to initial values, single-step errors may accumulate and increase over time, eventually causing the computation to diverge and preventing it from continuing. Second, the computed results of the PDE are affected not only by the reconstruction scheme, but also by various other factors in the solving algorithm. Therefore, reconstruction errors in the results may be evident or concealed in different test cases. Consequently, if a strict analysis of the difference in accuracy of the reconstruction scheme is required, specific solving algorithms should not be involved.

Considering the aforementioned factors, DOLINC and non-DOLINC schemes were used only for single-step reconstruction. The data of the mesh cells on which the reconstruction was based directly adopted the exact solution of the laminar Blasius equation (obtained using a high-resolution ODE solver). Different reconstruction schemes were applied to obtain face values, which were then compared with the Blasius solution at the face-center positions. Figs. 5 and 6 illustrate the reconstruction results on a boundary

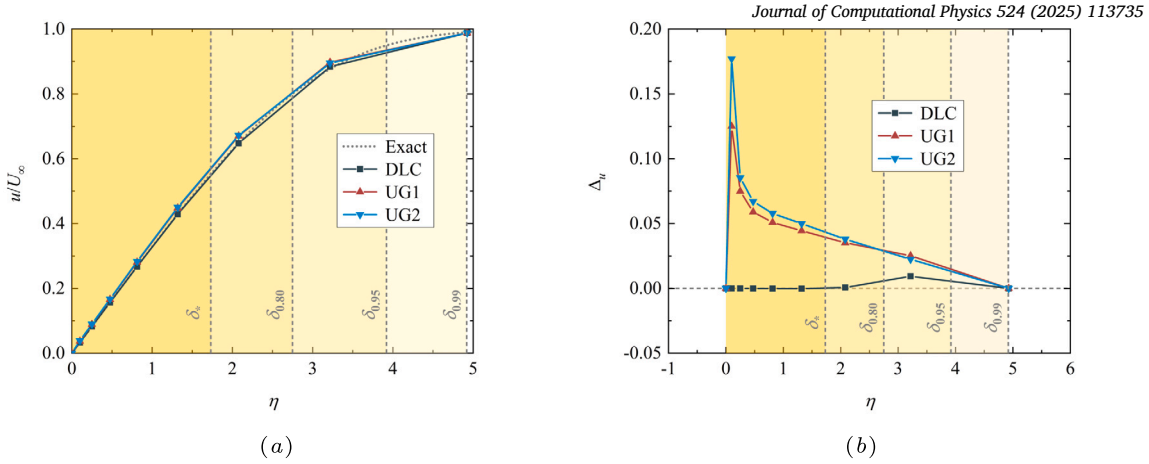

Fig. 5. Streamwise velocity component of mesh face obtained through single-step reconstruction and the exact solution. (*a*) Absolute values of dimensionless velocity, (*b*) relative error of dimensionless velocity compared to the exact solution.

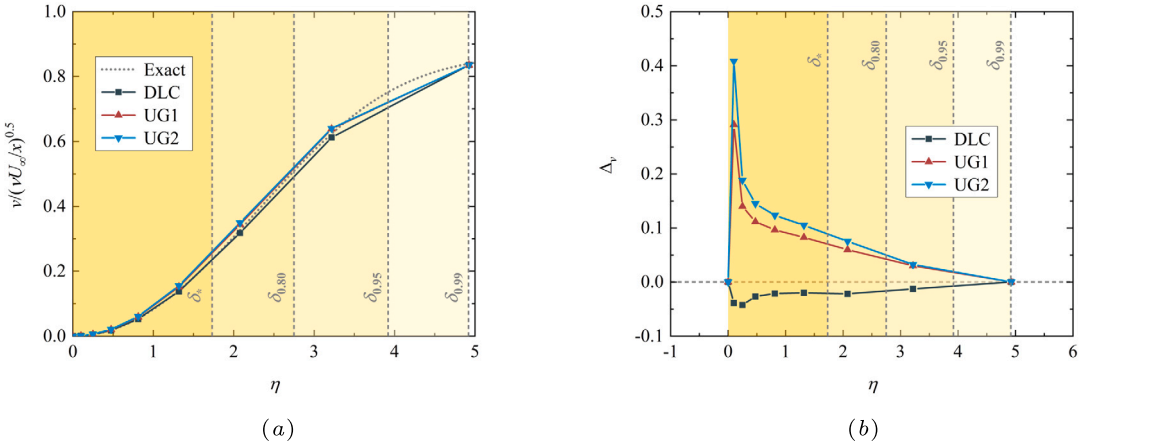

Fig. 6. Wall-normal velocity component of mesh face obtained through single-step reconstruction and the exact solution. (*a*) Absolute values of dimensionless velocity, (*b*) relative error of dimensionless velocity compared to the exact solution.

layer mesh with a vertical wall growth rate of 1.5 based on the USR3 scheme. Different methods were employed to implement the same high-order scheme: DLC represents USR3 implemented based on the DOLINC method, whereas UG1 and UG2 represent USR3 implemented based on the uniform-mesh-based gradient-approximation method. The difference between UG1 and UG2 is the specific gradient-calculation method, where the former considers the non-uniform characteristics of the mesh in the gradient calculation, resulting in more accurate gradient data than the latter.

Figs. 5 and 6 show significant differences between the DLC and UG results, regardless of whether the velocity component is reconstructed parallel or perpendicular to the wall. In Figs. 5(*b*) and 6(*b*), the relative errors of the velocity reconstructed by the DOLINC scheme are consistently low throughout the entire boundary layer, whereas both UG methods exhibit large reconstruction/interpolation errors in the strong shear region within the boundary layer. This accuracy difference in the single-step reconstruction accumulates as the computation progresses and ultimately has a significant impact on the computed results. Although the absolute values presented in the plots in Figs. 5(*a*) and 6(*a*) may seem similar, the distribution pattern of the velocities within the boundary layer fundamentally changed because of the errors introduced by the UG method. Comparing the results of the two velocity components, the differences between the DOLINC and UG methods in the wall-normal direction were more pronounced, as shown in Fig. 6. Therefore, if the computed boundary layer flow is still in a developing state, the uniform-mesh gradient-approximation method leads to larger errors, resulting in a less accurate distribution of the boundary layer velocities. Notably, UG1 and UG2, which used two different gradient-calculation methods, exhibited very small differences. The use of a more accurate gradient in UG1 did not effectively reduce the impact of the results produced by the UG method. In fact, at some locations, the relative error of UG1 surpassed that of UG2. Therefore, non-DOLINC methods cannot eliminate errors in non-uniform grids solely by improving gradient estimates; they can only marginally alter the reconstruction results and cannot guarantee that the altered results will always be more accurate.

The superiority of the DOLINC method, as shown in Figs. 5 and 6 is not specific to a particular grid growth rate or high-order reconstruction. We further investigated the global errors for different grid growth rates and different-order reconstruction schemes, as shown in Figs. 7 and 8. The errors in the figures were calculated using the L2 norm. Overall, the global errors in the computed

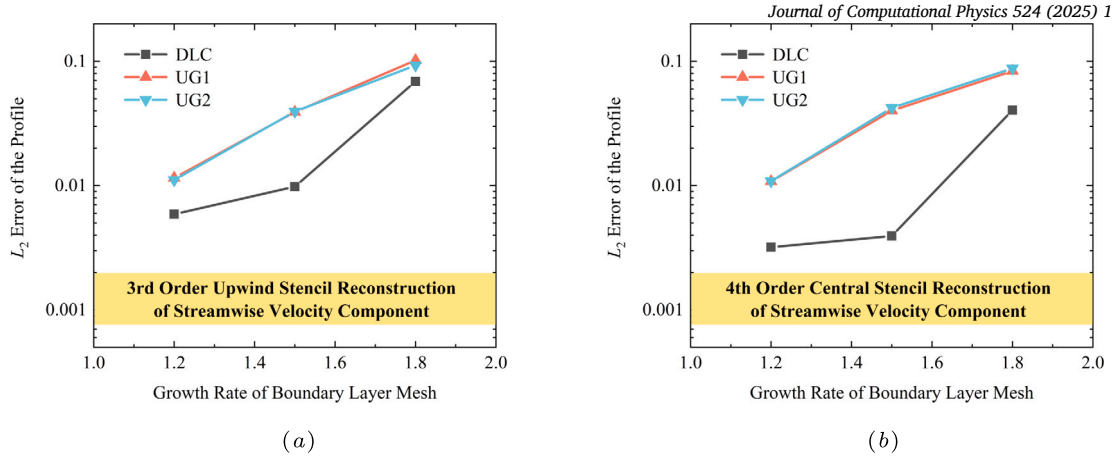

Fig. 7. Global errors of the streamwise velocity component with respect to mesh growth rate. (*a*) Third-order fixed-stencil reconstruction, (*b*) fourth-order fixed-stencil reconstruction.

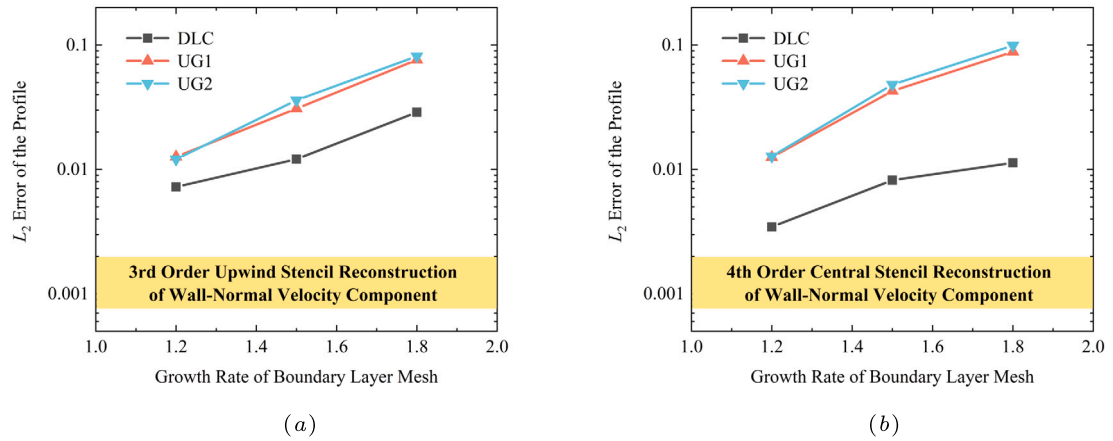

Fig. 8. Global errors of the wall-normal velocity component with respect to mesh growth rate. (*a*) Third-order fixed-stencil reconstruction, (*b*) fourth-order fixed-stencil reconstruction.

results increased with the grid growth rate. This is because, when the range of the boundary layer is fixed and the height of the first layer of the grid remains constant, a large grid-growth rate represents fewer grid cells. For different growth rates, the DOLINC scheme exhibited significantly lower global errors in reconstructing the wall-parallel and wall-normal velocity components compared with the non-DOLINC schemes. Notably, the global errors of UG1 and UG2 were almost identical, again indicating that the difference in the gradient-calculation approaches had minimal impact on the errors. However, the DOLINC method, which was based on increasing the reconstruction accuracy from third-order to fourth-order precision, showed a considerable reduction in global errors. This implies that the DOLINC method accurately achieved high-order precision in non-uniform grids. In contrast, this precision improvement was not observed when higher-order reconstruction schemes were implemented using the UG method; the error level of USR3-UG was consistent with that of CSR4-UG. This indicates that the main error of the UG method far surpassed the error of the interpolation polynomial itself, and the error in the UG scheme was primarily due to the error in the UG method rather than the error in USR3. Therefore, even if USR3 is upgraded to CSR4, the precision of the UG scheme cannot be enhanced because the error in the UG method itself does not improve. Considering that the inherent precision of USR3 is only third-order, UG-USR3 may have decayed to second-order or even lower precision.

3.2. Linear advection problem

The linear advection problem is one of the simplest hyperbolic conservation systems, and its multi-dimensional equation is formulated as follows:

$$\partial_t u + a_j u_{,j} = 0, \tag{67}$$

where a_j denotes the constant propagation velocity. In the advection transport of this problem, the original waveform/distribution maintains its shape during propagation with a uniform and steady propagation velocity across the entire domain. In the computation of

an incompressible flow, pure heat convection and mass-advection transport, neglecting diffusion, generally satisfy the linear advection transport equation. Therefore, the performance of numerical schemes in solving the linear advection equation reflects their practical effectiveness in characterizing the generalized passive scalar transport results in the computations of incompressible flow problems. For the one-dimensional case, the linear advection equation is simplified to

$$\frac{\partial u}{\partial t} + a\frac{\partial u}{\partial x} = 0. \tag{68}$$

An unstructured mesh solver was employed to solve the initial-value problem of the one-dimensional linear advection equation. Periodic boundary conditions were applied on both sides, allowing the initial waveform/distribution to re-enter from one endpoint after leaving the other, which is referred to as one transport period. To explore the performance of the high-order reconstruction schemes on smooth functions, Fig. 9 presents the computed results for the initial waveform corresponding to a Gaussian function of $\sigma = 0.0667$ after five transport periods. The top-left schematic provides a reference for the impact of numerical dissipation and dispersion on the resulting waveform. Fig. 9(a) illustrates the differences among various central schemes during the smooth function transport computation. CSR4 and CSR6 represent the fourth- and sixth-order reconstruction schemes implemented using the DOLINC method, respectively. CD2 is a common second-order accuracy central-differencing scheme for second-order FVM. CB3 is a scheme in the official release of OpenFOAM that utilizes cubic polynomial interpolation. In Fig. 9(a), CD2 exhibits typical numerical dissipation and dispersion phenomena associated with the central schemes. After five computation cycles, not only were the extrema significantly reduced, but numerical oscillations were observed on both the upwind and downwind sides. When using the CB3 scheme based on higher-order interpolation polynomials, both the numerical dispersion and dissipation were noticeably improved. However, as CB3 itself does not achieve fourth-order accuracy in practice because of its non-DOLINC implementation, differences in accuracy were still observed compared with the fourth-order CSR4 belonging to the DOLINC schemes. In this problem, the high-order reconstructions CSR4 and CSR6 exhibit almost no apparent numerical dissipation or oscillations. Fig. 9(b) shows the results for several upwind schemes, including the first-order upwind scheme (FOU), second-order upwind scheme (SOU), third-order scheme USR3, and fifth-order scheme USR5 implemented through the DOLINC approach. Excessively strong numerical dissipation were observed in the FOU results in a complete deviation of the transport outcome from the initial waveform. Compared with the conventional SOU scheme, the DOLINC USR3 scheme demonstrates better performance. While it has lower numerical dissipation, it exhibits almost no numerical oscillations on the downwind side. Similar to the central schemes CSR4 and CSR6, USR5 maintained its initial distribution shape well after five periods with minimal dissipation and oscillations. Regardless of whether they were central or upwind schemes, even if the construction principles of the schemes were so simple that they were not inherently non-oscillation types, simply increasing the order of the method could significantly reduce numerical dissipation and dispersion in practice. For the two non-oscillation TVD schemes in Fig. 9(c), the results for the VanLeer and MUSCL limiters both showed significant peak dissipation, greatly affecting the overall accuracy of the results. This again demonstrated the major drawback of TVD schemes in terms of accuracy degradation at smooth extrema points, leading to an inability to capture some small-scale continuous flow structures.

Although the Gaussian distribution is theoretically smooth, the discretized physical field after FVM discretization is effectively a piecewise function. Therefore, a discontinuous step on the grid faces in the discretized space becomes apparent when the width of the Gaussian distribution decreases to a certain extent. This causes the results to exhibit characteristics similar to those of computations of discontinuous functions, namely exacerbated numerical dissipation and numerical dispersion. As shown in Fig. 10, when the half-width of the Gaussian function decreases to $\sigma = 0.0333$, the high-order schemes that do not exhibit numerical oscillations and dissipation in Fig. 10 noticeable peak decay and oscillations. As shown in Fig. 10(a), the oscillations and dissipation of CD2 and CB3 continued to worsen; at this point, CSR4 also showed dissipation and oscillations. Compared with CSR4, the higher-order central reconstruction, CSR6, exhibited marginal oscillations; however, it accurately characterized the initial distribution owing to its higher accuracy. Similarly, in Fig. 10(b), while USR3 was still superior to SOU, the dissipation and oscillations on both sides were enhanced. At this point, USR5 also began to exhibit slight dissipation and oscillations. However, compared with the lower-order USR3, its performance advantage remained evident. Overall, when the discontinuity characteristics of the physical field were strengthened, simply increasing the order of the schemes could effectively mitigate the dissipation/dispersion phenomena, resulting in significantly improved computational results. The results of the non-oscillation schemes shown in Fig. 10(c) show the TVD scheme with the VanLeer limiter and demonstrate the performances of the third-order ENO reconstruction (ENO3) and fifth-order WENO reconstruction (WENO5) implemented using the DOLINC method for this problem. Evidently, the accuracy-degradation problem of the TVD schemes was more severe near the discontinuous positions close to the peak. In contrast, ENO3 and WENO5 exhibited significantly smaller numerical dissipation, maintaining the original wave series while strictly ensuring non-oscillation characteristics.

The results of advective transport with a rectangular-wave initial distribution were compared between the ENO/WENO and TVD schemes to further investigate the performance of the different schemes in handling strong discontinuities, as shown in Figs. 11 and 12. The figures show that ENO3 and WENO5, both of which belong to the second-order DOLINC scheme, exhibited significantly higher numerical accuracy at the discontinuity than the TVD-VanLeer scheme, with WENO5 showing a more pronounced advantage. ENO2 and WENO3, which rely on second-order reconstructions in the template for handling the discontinuity area, have fewer obvious advantages than the TVD scheme. Thus, the ENO/WENO series methods only demonstrated significantly better characteristics than the TVD scheme in handling discontinuities when they reached a particular high order [39]. Therefore, on unstructured grids, methods such as the DOLINC approach, which can achieve arbitrarily high orders, are more valuable than implementing a specific third-order scheme. Furthermore, the ENO and WENO schemes exhibited better symmetry in both the upwind and downwind discontinuities compared with the TVD scheme. The VanLeer limiter used in this case satisfied the property of the symmetry $\psi(1/r) = \psi(r)/r$. Other TVD limiters that do not satisfy this relationship may exhibit a poorer symmetry. Therefore, if only the capture of flow-

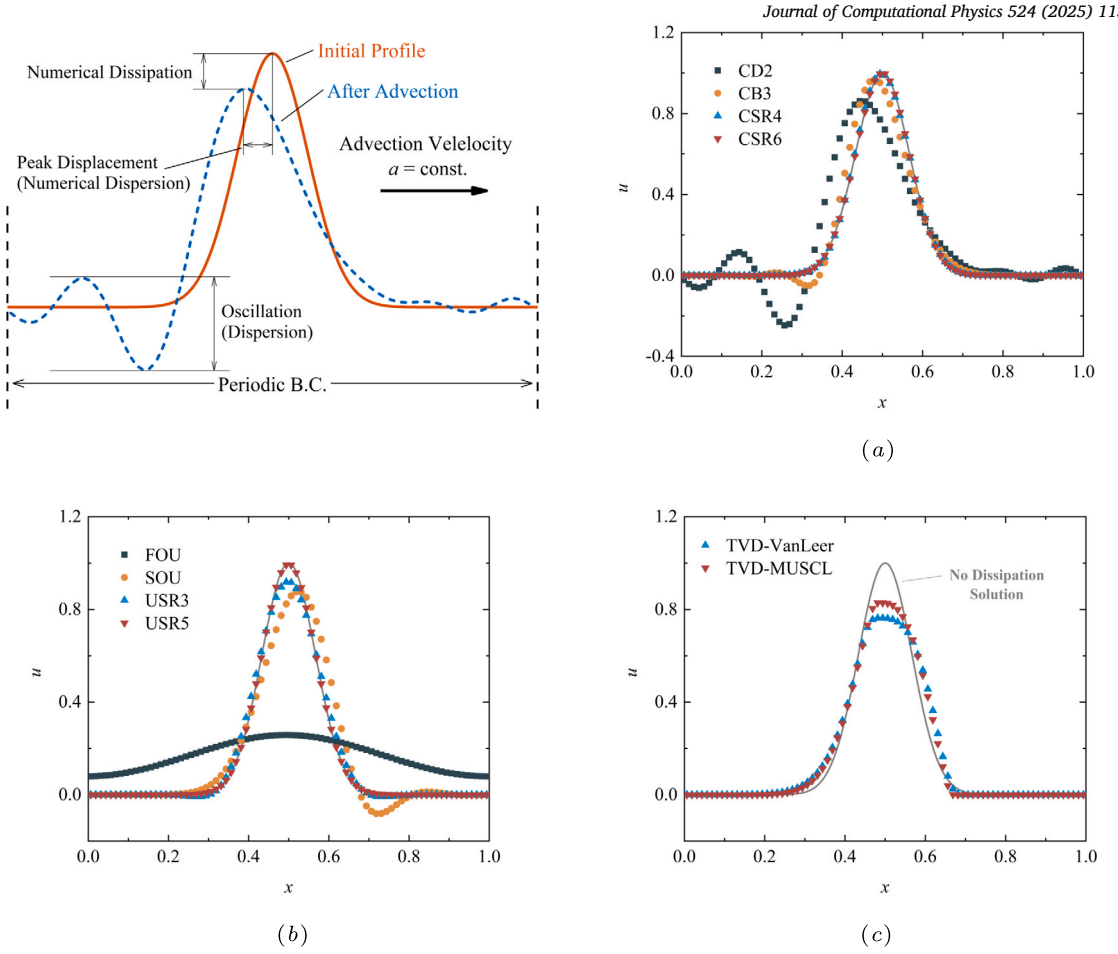

Fig. 9. Linear advection transport of a smooth Gaussian distribution. (*a*) Central schemes, (*b*) upwind schemes, (*c*) non-oscillatory schemes.

field discontinuities is considered, the TVD scheme performs well, and in terms of cost-effectiveness, industrial software using the TVD scheme has a clear advantage. However, when flow problems involve the capture of shockwaves and fine vortex structures simultaneously (such as typical supersonic turbulent flows), the most fatal issue manifested by the TVD scheme compared with higher-order schemes, such as ENO/WENO, is the previously mentioned accuracy degradation at the extremum points. Thus, the main drawback of the TVD scheme in industrial applications is its excessive dissipation of small-scale flow structures, leading to an inability to correctly handle fine vortex structures that may appear owing to instability in strong-discontinuity problems, rather than its handling of strong discontinuities. This is also a reason for caution when using the TVD scheme in LESs and DNSs for strong compressible flows.

To validate the actual convergence order of the DOLINC scheme, its global error was calculated based on an initial problem using a sine function. The initial errors of the results for each scheme with 10 grid cells were normalized to the same reference value to facilitate the comparison of the error-reduction rate with an increasing number of grid cells. At this point, the absolute-error size after refining the grid reflects the convergence speed of the scheme, as shown in Fig. 13. The original error data and corresponding convergence accuracy without normalization are presented in Tables 1 and 2, respectively. In the series of basic schemes shown in Fig. 13(*a*), commonly used low-order FVM schemes in industrial software, such as CD2 and SOU, and some special schemes, such as QUICK, all exhibit an actual convergence order of approximately two. The QUICK scheme, which is based on a quadratic interpolation polynomial derived from finite differencing, cannot achieve third-order theoretical accuracy when applied to the finite-volume method. In contrast, the fixed template schemes, USR3 and CSR4, implemented using the DOLINC scheme, achieved theoretical convergence accuracies of the third and fourth orders, respectively. In non-oscillation schemes, TVD schemes are limited by the first-order accuracy near the local extrema, resulting in a global error that does not exceed the second order. Therefore, the two TVD-limited schemes shown in Fig. 13(*b*) demonstrated orders of accuracy lower than two. In contrast, the two non-oscillation schemes, ENO3 and WENO5, implemented using the DOLINC method, achieved third- and fifth-order theoretical convergence accuracies, respectively. Notably, the convergence order of WENO5 quickly converged from approximately 4.2 on coarse grids to approximately 5, as shown in Table 2.

In some validation cases, the differences between the high- and low-order FVM schemes are insignificant. This is not because the performances of the schemes are similar, but rather because the chosen problem is relatively mild. A frequently used two-

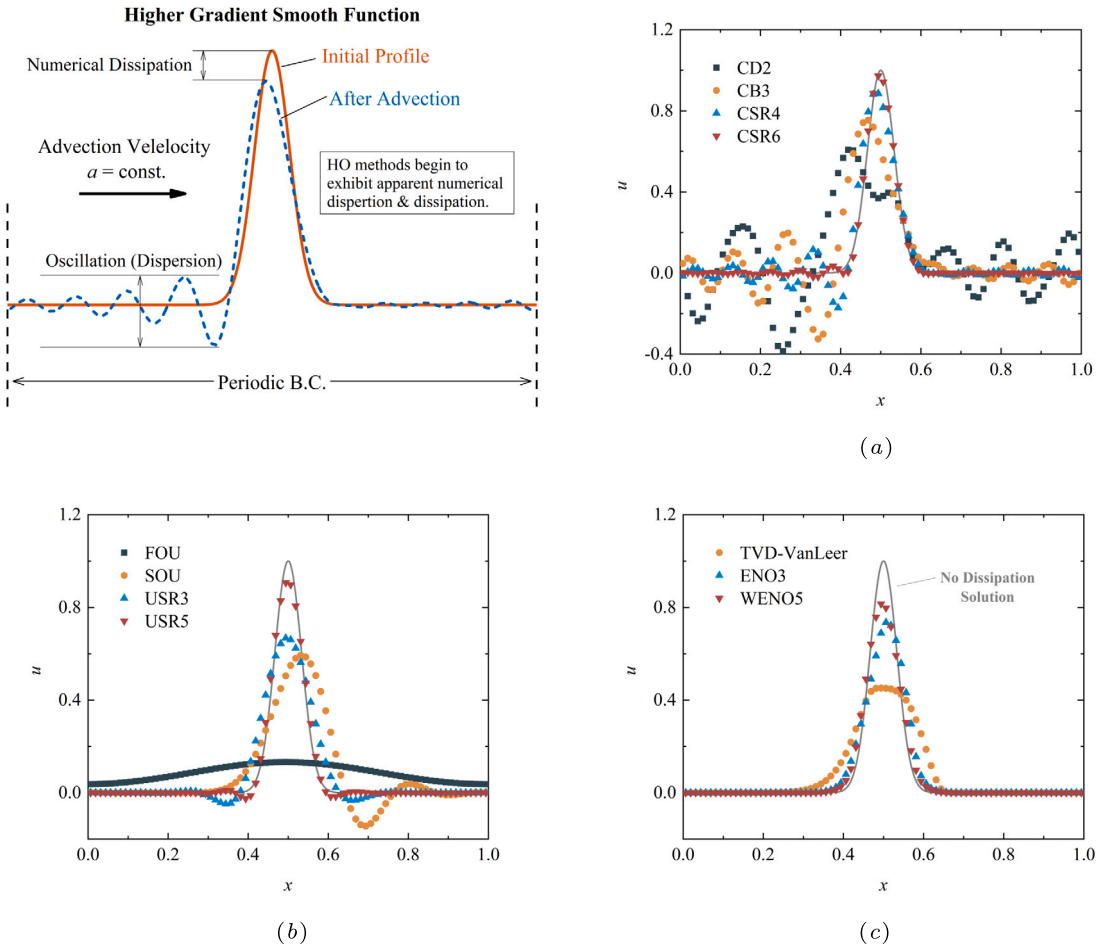

Fig. 10. Linear advection transport of a steep Gaussian distribution. (*a*) Central schemes, (*b*) upwind schemes, (*c*) non-oscillatory schemes.

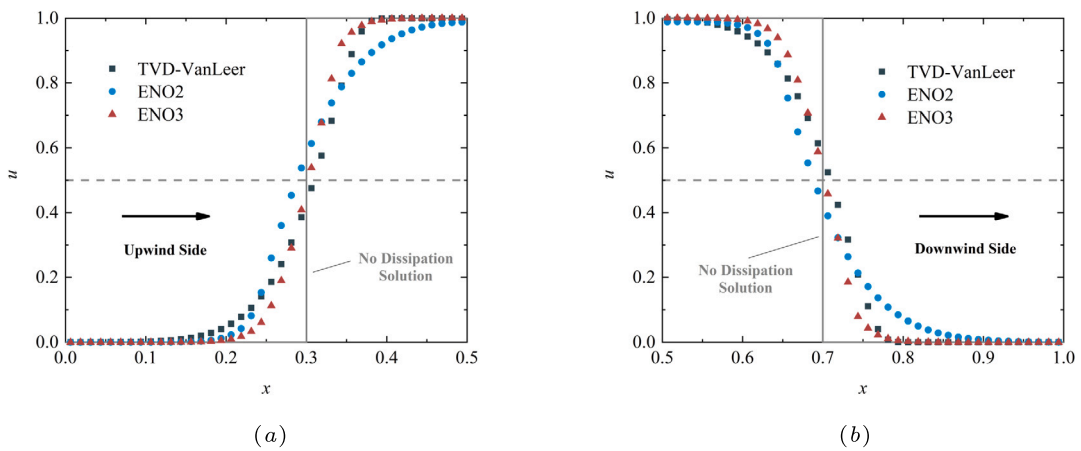

Fig. 11. Results after linear advection of the initial-step distribution calculated using the ENO method. (*a*) Upstream discontinuity on the windward side, (*b*) downstream discontinuity on the leeward side.

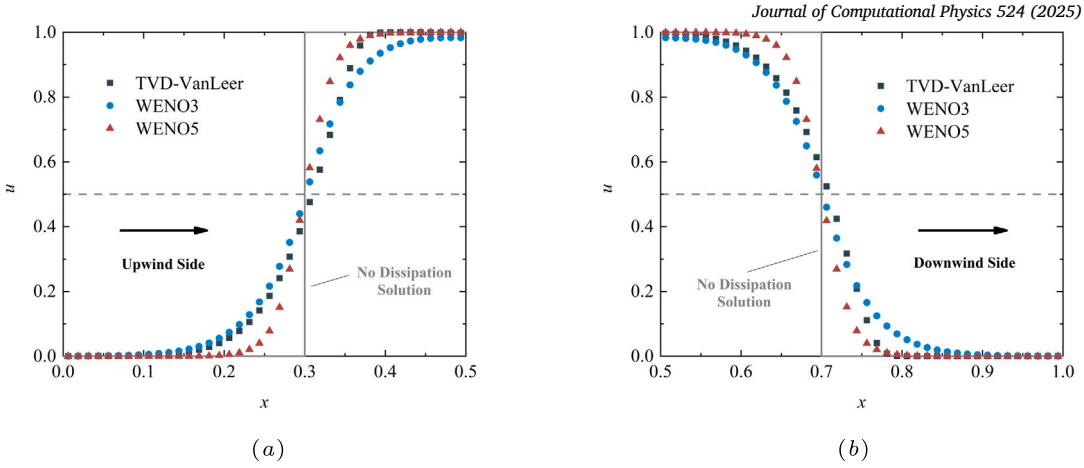

Fig. 12. Results after linear advection of the initial-step distribution calculated using the WENO method. (*a*) Upstream discontinuity on the windward side, (*b*) downstream discontinuity on the leeward side.

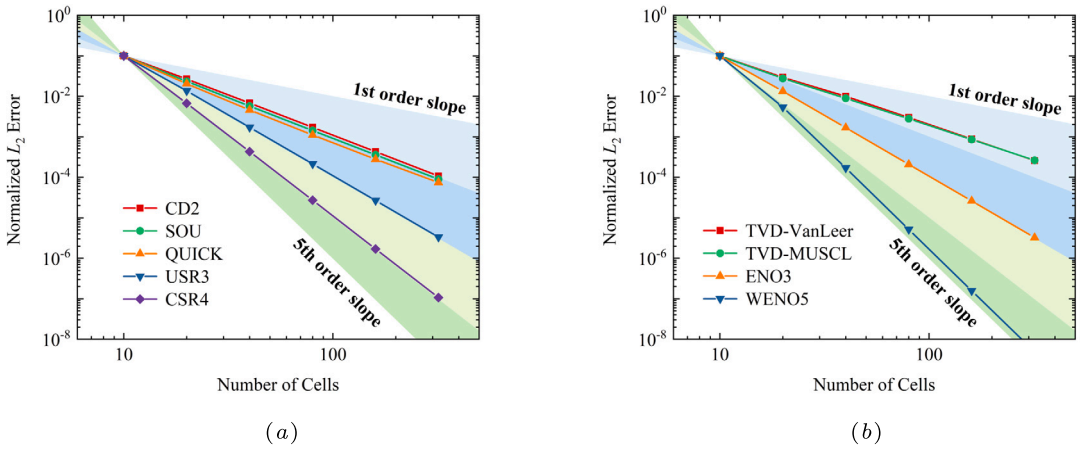

Fig. 13. Error convergence order of accuracy for different schemes. (*a*) Primitive scheme, (*b*) non-oscillatory scheme.

Table 1
L_2 error and convergence order for the primitive schemes.

Mesh	CD2		SOU		USR3		CSR4	
	Error	Order	Error	Order	Error	Order	Error	Order
10	0.264391		0.15908		0.084241		0.020641	
20	0.070505	1.91	0.03729	2.09	0.011493	2.87	0.001384	3.90
40	0.017996	1.97	0.009122	2.03	0.001446	2.99	8.87E-05	3.96
80	0.004536	1.99	0.002276	2.00	0.00018	3.00	5.59E-06	3.99
160	0.001138	1.99	0.00057	2.00	2.25E-05	3.00	3.51E-07	3.99
320	0.000285	2.00	0.000143	2.00	2.81E-06	3.00	2.21E-08	3.99

Table 2
L_2 error and convergence order for the non-ocsillatory schemes.

Mesh	TVD-VanLeer		TVD-MUSCL		ENO3		WENO5	
	Error	Order	Error	Order	Error	Order	Error	Order
10	0.250299		0.176635		0.086452		0.031202	
20	0.073412	1.77	0.049033	1.85	0.011572	2.90	0.001682	4.21
40	0.025002	1.55	0.01582	1.63	0.001449	3.00	5.30E-05	4.99
80	0.007493	1.74	0.004935	1.68	0.00018	3.01	1.61E-06	5.04
160	0.002205	1.76	0.001516	1.70	2.25E-05	3.00	4.93E-08	5.03
320	0.000647	1.77	0.000463	1.71	2.81E-06	3.00	1.53E-09	5.01

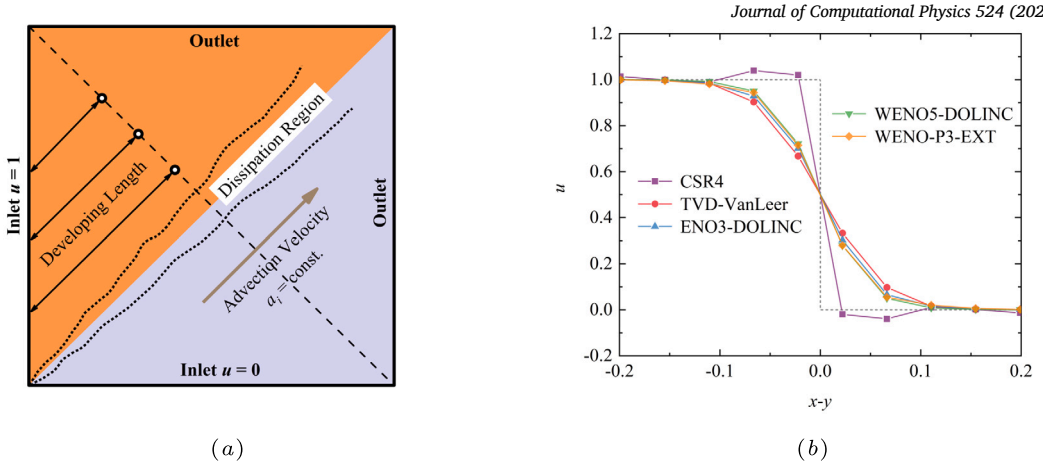

Fig. 14. Two-dimensional linear advection case. (*a*) Computational domain and boundary conditions setup, (*b*) enlarged view of the computed results along the dashed diagonal line.

dimensional convection transport problem for comparing scheme differences in industrial CFD software is shown in Fig. 14(*a*). It computes a pure advection problem within a square region. The advantage of this case is its simple geometry and boundary settings, which make it convenient to demonstrate some characteristics of the numerical schemes visually. Quantitative comparisons of the numerical dissipation can also be made by comparing the diagonal distribution perpendicular to the flow direction. Notably, this problem eventually reaches a steady state. Additionally, because the computational domain itself is not large, and the diagonal used for quantitative analysis is relatively close to the inlet boundary, the influence of numerical dissipation in this problem is relatively weak. Unlike the one-dimensional convection problem discussed earlier, where the dissipation becomes more significant as the computation time progresses, the numerical dissipation in this two-dimensional case, after reaching a steady state, is constrained within a certain range. Even at the most dissipative position in the upper-right corner, the experienced dissipation path is limited to a diagonal length that is considerably less than the dissipation path scale achievable in multiple periods of the one-dimensional convection problem. Therefore, in this mild case, the differences in the computed results between the different schemes were not significant. As shown in Fig. 14(*b*), the advantages of the high-order schemes are not reflected well in this case. Even low-order FVM schemes yield satisfactory results for this problem. The results shown in Fig. 14(*b*) represent an enlarged distribution in the central region, and the actual total length of the diagonal was 2.0. In addition to the ENO3/WENO5 schemes implemented based on the DOLINC method, this case also used the WENO-P3 scheme based on the unstructured-mesh k-exact method for computation. WENO-P3 also uses an interpolation polynomial for weighting and exhibits accuracy similar to that of WENO5 (supported by both Fig. 14 and Section 3.4.4). The specific implementation of the unstructured-grid WENO methods can be found in the literature [34]. To distinguish between the two WENO schemes, they are referred to as WENO5-DOLINC and WENO-P3-EXT. The figure shows that the resulting curves of the WENO5-DOLINC and WENO-P3-EXT basically overlapped, and no significant difference in accuracy was observed. Except for CSR4, which marginally overpredicted owing to numerical dispersion at the discontinuity, the other numerical schemes did not exhibit numerical oscillations.

The proposed DOLINC approach is suitable for high-order reconstructions on unstructured polyhedral mesh of various shapes. However, the inversion formulae on which the method is based are derived for non-uniform quadrilateral/hexahedral meshes. Therefore, the resulting DOLINC scheme is unconditionally accurate, achieving the corresponding theoretical order for these mesh shapes. Accuracy loss may occur to a certain extent when applied to grids of other shapes, such as triangular/tetrahedral meshes. To analyze whether the DOLINC scheme still retains its advantages in accuracy and efficiency over conventional second-order FVM schemes in the event of an accuracy loss, a two-dimensional linear advection was calculated using two kings of planar triangular mesh, as shown in Fig. 15. The first one consists of right-triangle cells that based on original quadrilateral cells and the second one was generated using normal-triangle cells. This case used the same boundary and initial conditions, as shown in Fig. 14. Moreover, the performance of different DOLINC schemes in linear advection computations was compared.

Fig. 16 shows the results obtained using the right-angled triangular mesh $32 \times 32 \times 2$ illustrated by Fig. 15(*a*). As shown in the Fig. 16, due to the mesh surfaces aligning precisely with the discontinuity, even with only half the grid density of the corresponding quadrilateral mesh, all schemes in Fig. 16 effectively captured the discontinuity. In Fig. 16(*b*), all non-oscillatory schemes captured the discontinuity within a single grid cell, demonstrating high accuracy compared to that of Fig. 14(*b*). Although various upwind schemes still exhibited some dispersion in Fig. 16(*a*), the overshoot magnitude was relatively small.

Fig. 17 compares several basic FVM schemes on normal triangular and quadrilateral meshes. The triangular mesh maintains nearly the same number of grid cells as the quadrilateral mesh. The figure shows that even conventional second-order FVM schemes produce noticeable differences when applied to quadrilateral and triangular meshes. Interestingly, the results on the triangular mesh are not always degraded; for some schemes, dissipation and dispersion were weaker on the triangular mesh as shown in Fig. 17. It should be noted that all central schemes and high-order upwind schemes diverge on this triangular mesh due to the numerical oscillations, so their results are not presented. Fig. 18 illustrates the performance of upwind and non-oscillatory schemes on this triangular

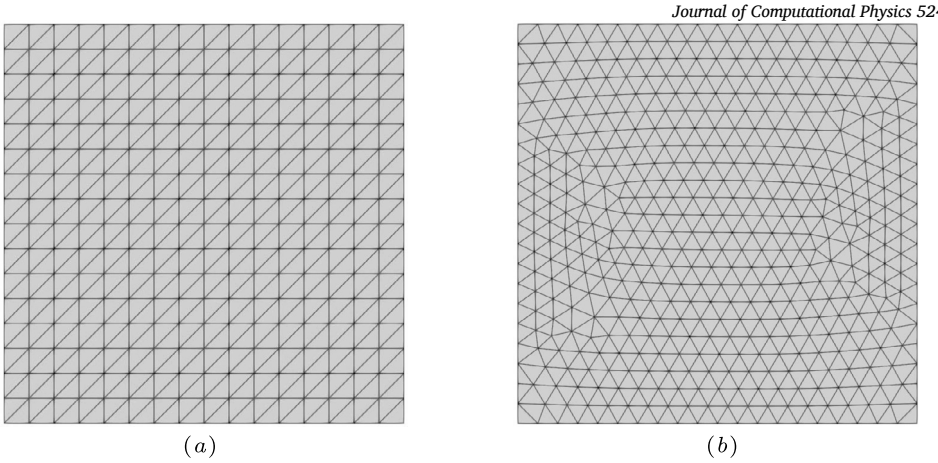

Fig. 15. Schematic of two-dimensional triangular mesh for the linear advection problem. (*a*) Right-triangle cells, (*b*) normal-triangle cells.

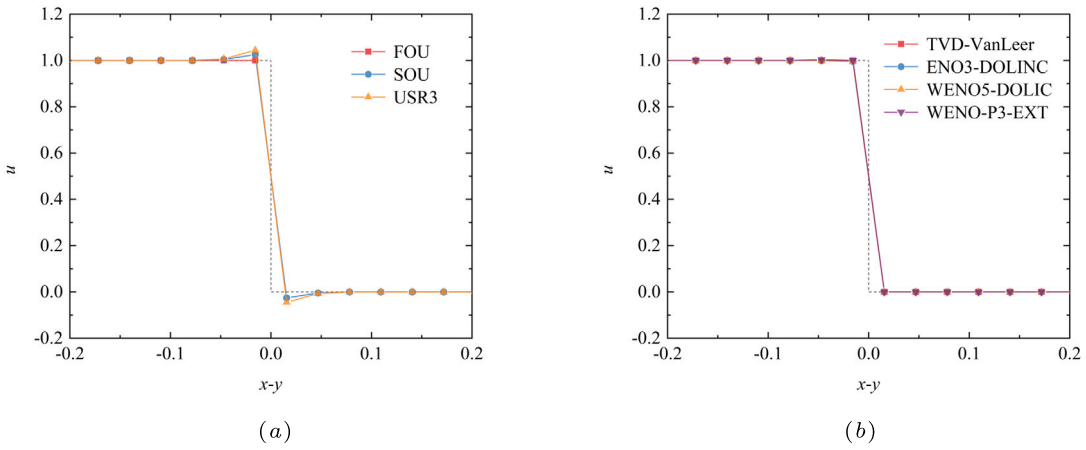

Fig. 16. Results of two-dimensional linear advection transport on a right-triangle mesh. (*a*) Upwind schemes, (*b*) non-oscillatory schemes.

mesh. Fig. 18(*a*) shows that the USR3 scheme displayed results similar to the SOU scheme. For the non-oscillatory schemes shown in Fig. 18(*b*), the results on triangular meshes were similar to those on quadrilateral meshes, as shown in Fig. 14(*b*). The DOLINC-based ENO3 and WENO5 schemes exhibited numerical dissipation similar to the TVD scheme using the VanLeer limiter, effectively capturing discontinuities without numerical oscillations. The WENO-P3-EXT scheme based on the *k*-exact method demonstrated higher computational accuracy and enhanced discontinuity capture. This observation differs from the results observed in Fig. 14(*b*). Due to the inexact inversion formula when handling triangular cells, the DOLINC schemes exhibited lower accuracy than the traditional *k*-exact WENO method. This WENO method seems more suitable on triangular grids because of its adaptive multi-dimensional stencils. However, a significantly higher computational cost is incurred (shown below).

In addition to verifying the accuracy of the DOLINC scheme, tests were conducted to assess its computational efficiency. Based on the linear advection equation, we compared the computational-time differences between the native schemes provided in the OpenFOAM release and high-order schemes implemented using the DOLINC method in one-dimensional and two-dimensional cases. The results are listed in Tables 3 and 4. In Table 3, we show the increase in computational cost when improving the accuracy with different schemes using the second-order central differencing scheme CD2 as the baseline. CD2 was chosen as the baseline because, in low-order FVM industrial software, when conducting incompressible turbulent LES or DNS simulations, CD2 is often recommended to avoid excessive numerical dissipation, even if numerical oscillations may occur. Owing to the lack of higher-accuracy high-order schemes, the only strategy for reducing errors and improving accuracy in the second-order FVM is to the continuous refinement of the grid. When the number of grid cells was increased to twice the baseline value (while keeping the CFL number constant), as shown in Table 3, the computational time increased by a factor of 1.5 times the original (an increment of 49%). By further refining the grid to four times the baseline results in a nearly 170% increase in computational time, the accuracy of CD2 approached that of CSR4. Choosing other high-order schemes while keeping the grid resolution constant yielded a clear advantage in terms of computational efficiency compared to simply refining the grid. Even for the most time-consuming WENO5-DOLINC scheme, the time only increased by 44%. The basic schemes, USR and CSR, implemented using the DOLINC method, exhibited computational times approximately equivalent to those of the native SOU scheme without a significant increase owing to the use of high-order schemes. Considering that the DOLINC method can further improve the efficiency of the actual code-implementation algorithm, USR3 and CSR4 had shorter

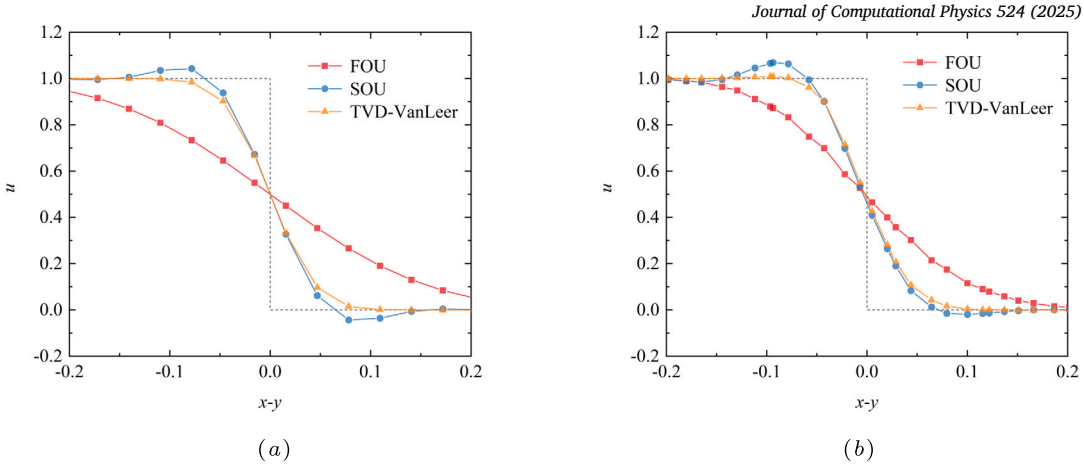

Fig. 17. Comparison of two-dimensional linear advection transport between basic FVM schemes. (*a*) Quadrilateral mesh, (*b*) normal-triangle mesh.

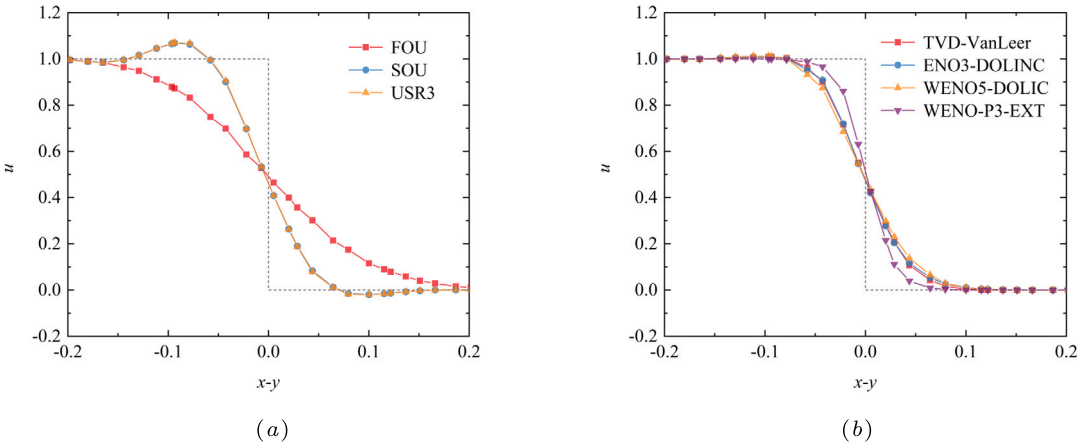

Fig. 18. Results of two-dimensional linear advection transport on a normal-triangle mesh. (*a*) Upwind schemes, (*b*) non-oscillatory schemes.

Table 3
Compuational-time increment using different schemes for a one-dimensional case (with CD2 as the reference).

Primitive Schemes	CPU Time Increase	DOLINC Schemes	CPU Time Increase
CD2	Baseline	USR3	0.16
CD2 (2x Refinement)	0.49	USR5	0.29
CD2 (4x Refinement)	1.69	CSR4	0.18
CB3	0.42	CSR6	0.29
SOU	0.24	WENO5	0.44

computational times than the second-order upwind scheme. This is reflected by the native CB3 scheme, which also uses a third-order polynomial, requiring more computational time than the CSR4 scheme, which has better computational accuracy.

With the introduction of advanced methods such as ENO/WENO, the computational times of high-order schemes tend to increase compared to basic schemes. Table 4 shows a significant difference in the computational times between the CSR4-DOLINC and ENO/WENO-DOLINC. The additional computational cost was primarily due to the more complex implementation procedure of these non-oscillation methods, rather than being closely related to the DOLINC method. However, compared with the WENO-P3 scheme, the ENO/WENO schemes implemented using the DOLINC method demonstrated a significant speed advantage. The computational cost of WENO-P3 in the table is 50–90 times that of the TVD scheme, whereas that of WENO5 is approximately six times that of the TVD scheme. This gap continues to widen in terms of the computational time after grid refinement. Therefore, high-order schemes with similar accuracy implemented based on the DOLINC method exhibit higher computational efficiency compared to unstructured-grid high-order schemes implemented using multi-grid templates.

Regarding triangular mesh, despite the WENO-P3 scheme showing higher accuracy than WENO5-DOLINC, the unaffordable cost remained significant. As shown in Table 5, the DOLINC scheme required, on average, five times more computational time using

Table 4
Computation time of various schemes for a two-dimensional case (quadrilateral mesh) at different grid resolutions.

Schemes	CPU Time	
	Standard mesh	Refined mesh
TVD-VanLeer	8.05	50.49
CSR4-DOLINC	11.57	77.28
ENO3-DOLINC	31.57	261.35
WENO5-DOLINC	48.41	324.49
WENO-P3-EXT	429.72	4514.03

Table 5
Computation time of various schemes for a two-dimensional case (triangular mesh).

Schemes	CPU Time	
	Standard mesh	Refined mesh
TVD-VanLeer	11.56	80.21
ENO3-DOLINC	54.19	344.64
WENO5-DOLINC	73.78	536.67
WENO-P3-EXT	758.33	7625.72

the one of the TVD scheme as a reference, while the WENO3-P3 scheme increased time by more than 65 times. Compared to the WENO5-DOLINC scheme, the WENO-P3 scheme required at least 900% more computational time for only marginal improvements in accuracy, making it inappropriate from a computational efficiency perspective. Therefore, further studies for developing a more accurate DOLINC inversion accurately describing triangular/tetrahedron grid behavior are essential, given its high computational efficiency.

3.3. Discontinuities in the Burgers equation

The Burgers equation is another typical hyperbolic conservation system that is commonly employed to study the characteristics of shocks in fluid dynamics. The multi-dimensional form of the Burgers equation is given by:

$$\partial_t u_i + u_j u_{i,j} = 0. \tag{69}$$

The Burgers equation is used to assess the performance of the spatial schemes in handling the generation and propagation of discontinuities. In one dimension, the corresponding governing equation is as follows:

$$\frac{\partial u}{\partial t} + u\frac{\partial u}{\partial x} = \frac{\partial u}{\partial t} + \frac{\partial}{\partial x}\left(\frac{1}{2}u^2\right) = 0 \tag{70}$$

The first solved problem involves the propagation of a discontinuity initially located at $x = -0.5$ m with a propagation speed of 0.2 m/s, constituting a Riemann problem for the Burgers equation. Fig. 19 shows the propagation results at $t = 5$ s obtained using different schemes. The results from the straightforward Lax–Friedrichs (LF) scheme exhibited significant numerical dissipation, leading to a thickness of six grid cells for the discontinuity after 5 s. By incorporating the WENO5-DOLINC reconstruction into the LF scheme, replacing the original method that directly used cell averages for computation, the obtained WENO5-LF scheme significantly improved the numerical dissipation of the LF scheme and enhanced the accuracy of discontinuity capture. Thus, the WENO5-LF scheme achieved a good balance between computational speed and accuracy. Combining the WENO5 reconstruction with a more accurate approximate Riemann solver yielded better performance. For example, as shown in the figure, the results from the WENO5-Roe scheme demonstrate a further improvement in accuracy compared to the WENO5-LF scheme. The waveform obtained by WENO5-Roe closely matched the theoretical discontinuity, and the error was confined to one grid cell on either side of the discontinuity.

The second problem solved employed a sine function as the initial field to study the discontinuity-generation process in a smooth field. The computational results are presented in Fig. 20, where the dashed lines represent the initial field distributions. The results for different grid resolutions and various orders of WENO methods were compared. The WENO5 scheme, implemented based on the DOLINC method, accurately captured the generated discontinuity even on coarse grids and exhibited an accuracy comparable to that of finer grids. The lower-order WENO3 scheme performed well on finer grids when coupled with the solver, but errors near the discontinuity increased marginally on coarser grids compared to the precise results obtained with WENO5. The non-oscillatory finite-volume scheme implemented using the DOLINC method on unstructured grids showed that increasing the order of the scheme can achieve high-precision results with a lower grid density, highlighting the accuracy of the DOLINC method in the inversion calculation.

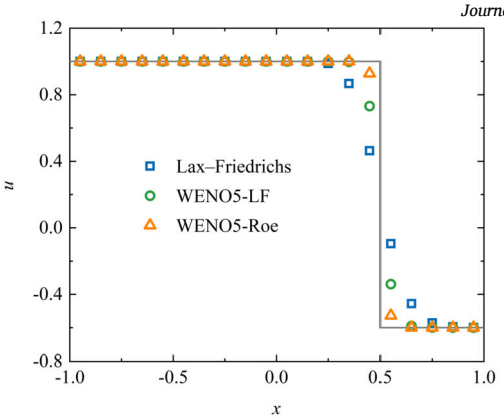

Fig. 19. Propagation results of the pre-existing discontinuity problem.

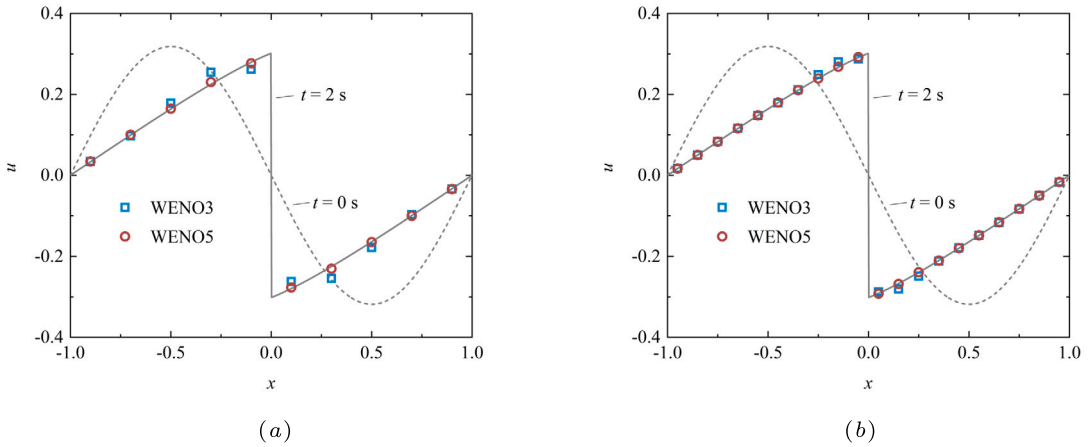

Fig. 20. Results of discontinuity generation in a smooth initial field under different grid resolutions. (*a*) Coarse gird and (*b*) fine grid.

3.4. Solving the Euler equations

Finally, we apply the DOLINC scheme to solve the inviscid-fluid governing equations, known as the Euler equations. The Euler equations for hyperbolic conservation systems are as follows:

$$\begin{aligned} \partial_t \rho + (\rho u_j)_{,j} &= 0, \\ \partial_t (\rho u_i) + (\rho u_i u_j + p\delta_{ij})_{,j} &= 0, \\ \partial_t (\rho E) + (\rho u_j E + u_j p)_{,j} &= 0. \end{aligned} \tag{71}$$

x-split multi-dimensional Euler equations have the following form:

$$\begin{aligned} \frac{\partial \rho}{\partial t} + \frac{\partial \rho u}{\partial x} &= 0, \\ \frac{\partial \rho u}{\partial t} + \frac{\partial \rho u u}{\partial x} + \frac{\partial p}{\partial x} &= 0, \\ \frac{\partial \rho v}{\partial t} + \frac{\partial \rho u v}{\partial x} &= 0, \\ \frac{\partial \rho w}{\partial t} + \frac{\partial \rho u w}{\partial x} &= 0, \\ \frac{\partial \rho E}{\partial t} + \frac{\partial (\rho E + p) u}{\partial x} &= 0. \end{aligned} \tag{72}$$

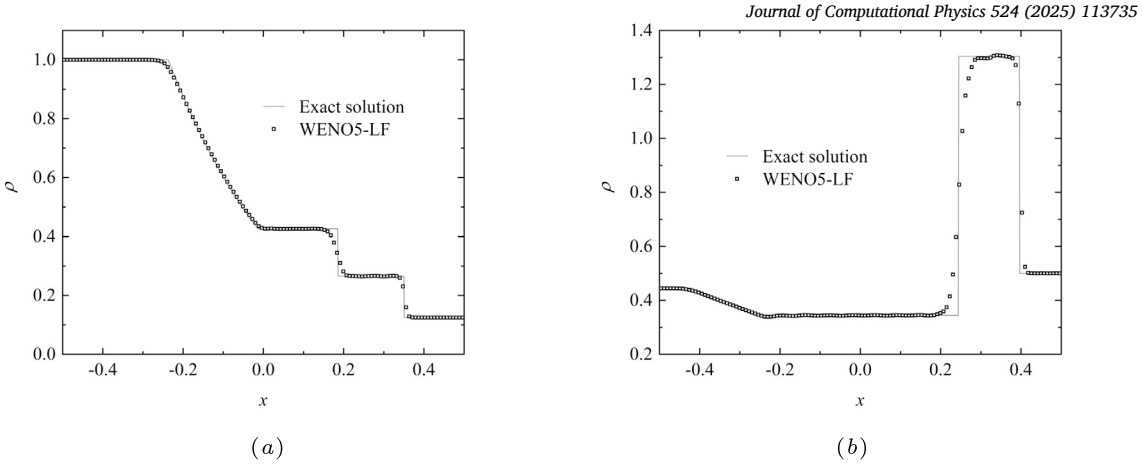

Fig. 21. Density of the one-dimensional Riemann problem. (*a*) Sod problem at 0.20 s, (*b*) Lax problem at 0.16 s.

Euler equations for one-dimensional problems can be simplified to

$$\begin{aligned} &\frac{\partial \rho}{\partial t} + \frac{\partial}{\partial x}(\rho u) = 0, \\ &\frac{\partial \rho u}{\partial t} + \frac{\partial}{\partial x}(\rho u^2 + p) = 0, \\ &\frac{\partial \rho E}{\partial t} + \frac{\partial}{\partial x}(\rho u E + p u) = 0. \end{aligned} \tag{73}$$

To better demonstrate the improvement in numerical accuracy achieved by the DOLINC scheme, the multi-dimensional Euler equations solvers discussed in this section do not employ approximate Riemann solvers, such as the HLLC or Roe schemes. Instead, they are combined with a more generally efficient LF scheme or local LF scheme [29,30]. In this study, the schemes used in the solver are termed "flux scheme" to distinguish them from the face interpolation/reconstruction scheme, the main focus of the proposed DOLINC approach. This treatment highlights the enhanced computational accuracy of the DOLINC method for low-order finite-volume methods on unstructured grids.

3.4.1. One-dimensional Riemann problem

Fig. 21 illustrates the computational results for two classical one-dimensional Riemann problems: the Sod [42] and Lax problems [31]. Overall, even with the increased numerical viscosity introduced by the LF scheme, the WENO5-LF Riemann problem solver maintained good accuracy. The captured expansion and shock waves in both the Sod and Lax problems appeared clean and clear in the plot, with only a marginal dissipation observed at the contact discontinuity.

A further comparison of the LF, TVD-LF, and WENO5-LF schemes in terms of computational details is presented in Fig. 22. In the capture of the left-propagating expansion wave in Fig. 22(*a*), the combination of the LF scheme with the WENO5 scheme noticeably reduced the numerical dissipation. In Fig. 22(*b*), which depicts the capture of the right-propagating contact discontinuity and shock wave, the LF scheme with WENO5 reconstruction achieved sharper discontinuity interfaces. The computational accuracy of the TVD scheme was marginally lower than that of the WENO5 scheme; however, overall, the accuracies remained very close, reaffirming that the shortcomings of the TVD scheme are not in capturing discontinuities in Riemann problems but rather in the previously discussed issue of extrema reduction.

3.4.2. Two-dimensional Riemann problem

The two-dimensional Riemann problem proposed by Schulz-Rinne et al. [37] has been widely used to validate the accuracy of numerical methods in studies related to high-order schemes and Riemann solvers [5,7,16]. However, this problem posed significant challenges for most industrial software packages; thus, achieving high-precision flow substructures using high-order methods is difficult. Although low-order FVM methods can accurately compute shockwaves/expansion waves developed in the flow field, they fail to correctly capture the vortex street generated by the Kelvin–Helmholtz instability owing to their substantial numerical dissipation. The performance difference of the TVD scheme mentioned earlier in capturing the flow-field discontinuities and small-scale flow structures is evident here. In Fig. 23, the TVD scheme (based on the VanAlbada limiter) significantly suppressed the Kelvin–Helmholtz instability. In the case with a lower resolution of 1024^2 grids, the TVD scheme almost eliminated all possible unstable vortex streets associated with the Mach stem and contact discontinuity. The flow characteristics included a mushroom cap and two-dimensional shockwaves. Even when the TVD scheme was applied with a resolution of 2048^2 grids, the vortex street did not form on either side of the contact discontinuity, and only the Mach stem exhibited marginal fluctuations. However, when the WENO5-DOLINC scheme was used for computation at a resolution of 1024^2 grids, it successfully captured the Kelvin–Helmholtz instability of both the Mach stem and contact discontinuity. The coherent vortex structures on the coarse grid with the WENO5 scheme were clearly visible and

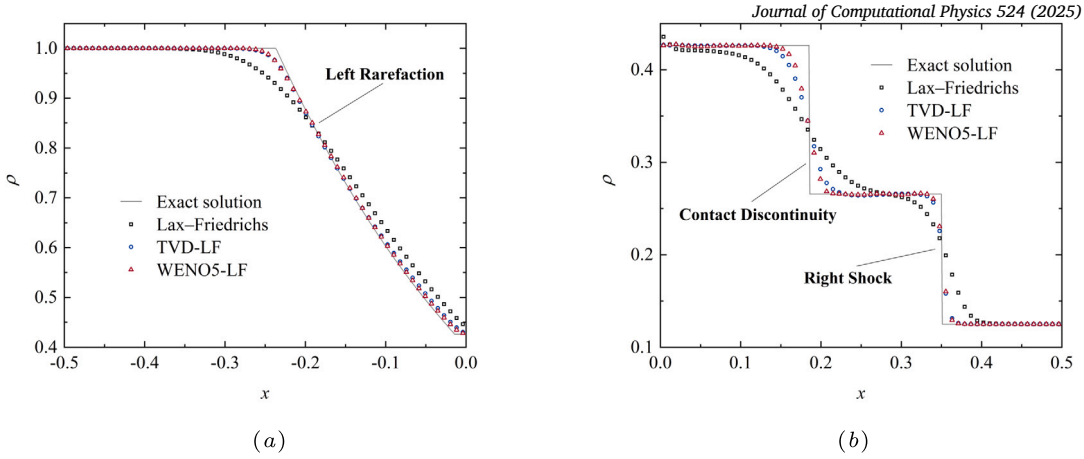

Fig. 22. Results of the Sod problem (one-dimensional shock tube) using different schemes for the Riemann problem. (*a*) Expansion wave on the left, (*b*) shock and contact discontinuity on the right.

surpassed the results achieved by the TVD scheme on a finer grid of 2048^2 grids. As the WENO5 scheme was further increased to a resolution of 2048^2 grids, the flow details induced by the Kelvin–Helmholtz instability became clearer, the unstable position of the contact discontinuity advanced, and vortices emerged within the mushroom cap. At this point, the capture results of the TVD scheme with 2048^2 grids could not compete with those achieved by the WENO5 scheme with the same resolution.

High-order schemes implemented based on the DOLINC method also have an advantage over the TVD scheme in terms of efficiency in improving computational accuracy. When 64 processors were used for parallel computation, the CPU time for the TVD scheme with a resolution of 1024^2 grids was 1290.28 s. Using this as a benchmark for accuracy and speed, only approximate capture results were obtained. When the grid was refined to a resolution of 2048^2 for more flow details, the computation time increased to 15599.9 s, nearly 11.09 times the baseline cost for additional accuracy. However, when the resolution was maintained at 1024^2 grids and the high-order WENO5 scheme was chosen for computation, the total time was only 4880.49 s, increasing the accuracy at an additional cost of only 2.78 times the baseline. As discussed in the analysis of Fig. 23, the coarse-grid WENO5 not only surpassed fine-grid TVD in terms of accuracy, but also consumed only 31% of the computation time of the latter, reducing the cost of accuracy improvement by almost 75%. This again demonstrated that high-order schemes for unstructured grids based on the DOLINC method had better computational efficiency than the existing schemes in conventional low-order FVM architectures. The advantages of DOLINC, in terms of accuracy and speed, along with its relatively simple integration, are expected to accelerate the transition from industrial software to high-order methods. Fig. 24 shows the computational results of the DOLINC scheme at an extreme grid resolution. The capturing of vortex street instability is comparable to the results obtained using high-order schemes in structured-grid high-precision solvers in previous research [16], reflecting the significant improvement in the accuracy limits achieved by the DOLINC method for low-order FVM methods.

3.4.3. Forward-facing step

The forward-facing step problem is another method for analyzing the accuracy of numerical schemes in capturing compressible flow features. This flow field involves multiple shock reflections and contact discontinuity phenomena. Similar to the two-dimensional Riemann problem, various discontinuities in the flow field can be effectively captured using the TVD scheme despite its high numerical dissipation. However, the dissipation in the TVD scheme suppresses instabilities induced by local velocity slip, preventing the calculation from capturing the continually developing and shedding vortex street downstream. To further study the performance of the DOLINC schemes in solving the Euler equations, the specific setup of Woodward and Colella [51] was employed to simulate the forward-facing step flow using both the TVD scheme with VanAlbada limiter and the DOLINC scheme of WENO5. Three mesh resolutions were selected, namely 600×200, 1200×400 and 2400×800. Figs. 25 and 26 illustrate the results.

Fig. 25 shows that for the forward-facing step problem, the behaviors based on different grid resolutions and interpolation/reconstruction schemes were very similar to those of the two-dimensional Riemann problem. The TVD and the WENO5- DOLINC schemes captured all shock waves and contact discontinuities. The contours of various discontinuity interfaces in the results were sharp and well-defined, and increasing the grid resolution reduced the computed discontinuity thickness. However, compared to the DOLINC scheme, the excessive numerical dissipation of the TVD scheme prevented the flow field from capturing any instability vortices. At a resolution of 1200×400, the flow field calculated with the TVD scheme showed no instability features. However, the WENO5 scheme exhibited characteristics of instability vortex development even at a resolution of 600×200. At 1200×400, the KH instability vortices induced by velocity slip downstream of the top normal shock were more observable. The vortex structures grew and shed progressively downstream with the flow. Compared to other WENO implementations in previous studies [15], the WENO5-DOLINC scheme based on the DOLINC approach achieved similar accuracy and effectively captured the instability vortex street. The current solver uses the simple local LF scheme for the flux scheme without any approximate Riemann solvers. However, the performance of the WENO5-DOLINC scheme could be improved by more accurate methods, such as the Roe or HLLC scheme.

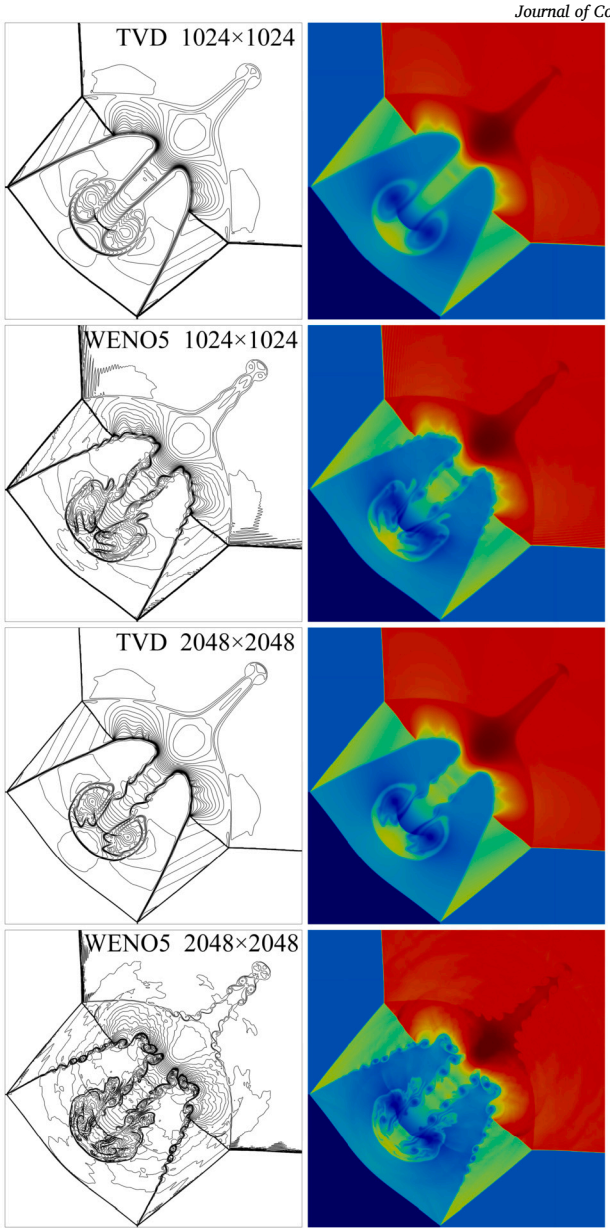

Fig. 23. Density of the two-dimensional Riemann problem at 1.1 *s* (contour lines on the left range from 0.15 to 1.7 with 31 lines; the colormap on the right ranges from 0.14 to 1.75).

In Fig. 26, the WENO5-DOLINC scheme combined with the LF flux scheme on a 2400×800 grid captured very detailed structures of instability vortex streets and wave family. Due to its higher resolution accuracy, the initiation of vortex structures at velocity slip appeared closer to the main shock wave, exhibiting strong unsteady characteristics. Since the solved equations are inviscid Euler equations, a convergent solution does not exist. Therefore, the far-upstream location where instability first occurs also indicates the high accuracy of the numerical scheme.

3.4.4. Rayleigh–Taylor instability

Another widely studied case in high-order methods research is the Rayleigh–Taylor instability problem [16,36,39]. In contrast to Riemann problems, the Rayleigh–Taylor instability is induced by gravitational effects on two fluids of different densities. This involves a lower overall Mach number and primarily focuses on the initiation and evolution of fine vortex structures. Because this problem has been extensively described in the literature, we did not go into repetitive details. Specific settings are available in the cited literature and are well-known. The initial perturbation at time zero was set as follows:

$$v = \varepsilon \cos(8\pi x) \sin^{\tau}(\pi y), \tag{74}$$

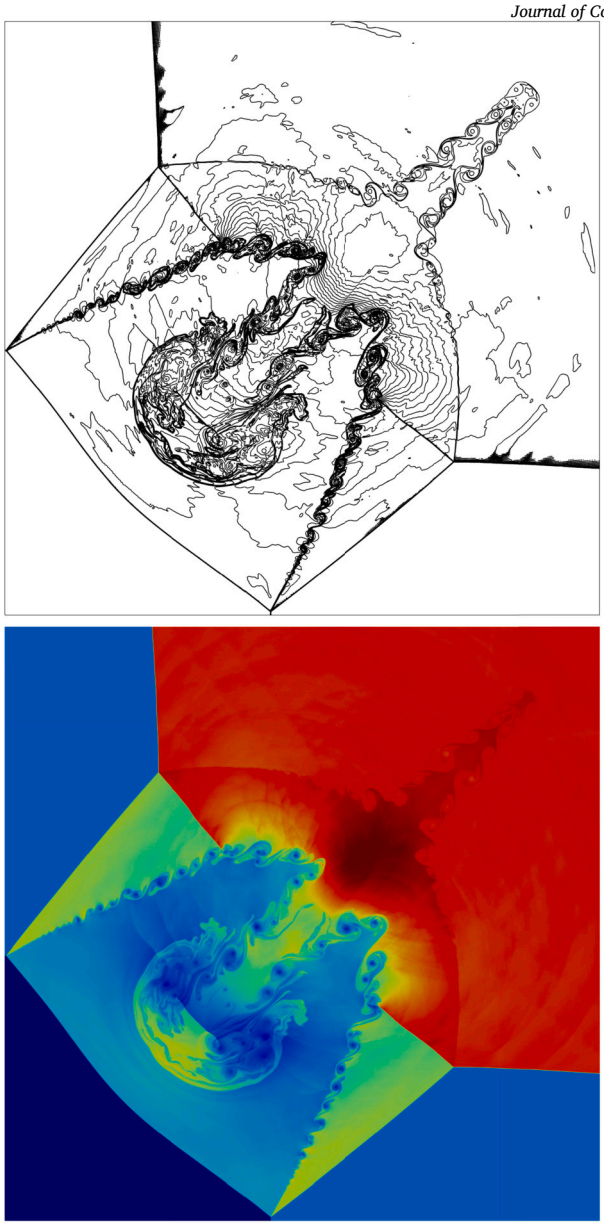

Fig. 24. Contour lines and colormap of density obtained through WENO5 scheme at the extreme resolution of 4096×4096.

where $\epsilon = \mathrm{Ma}_0 \times a_0$. We take $\tau = 6$, $a_0 = \sqrt{0.5\gamma}$, and $\mathrm{Ma}_0 = 0.1$. In contrast to the Riemann solvers used in other studies and LF-based Euler equation solver used in the other cases in this section, this case employs the native rhoPimpleFoam solver based on a pressure-coupling algorithm to showcase the effects of the combinations of different types of solvers with the DOLINC scheme. The calculations were carried out using the TVD scheme, WENO5 scheme implemented based on the DOLINC method, and WENO-P3 scheme based on the k-exact method and Gaussian integration, as shown in Figs. 27, 28 and 29.

In this low-Mach compressible flow problem, the absence of physical viscosity enabled the capture of finer vortex structures, indicating that the solution method possessed higher numerical accuracy and lower numerical viscosity. As shown in Fig. 27, at the four grid resolutions, the WENO5 scheme captured more flow substructures than the TVD scheme. The difference in capturing the flow details increased with the continuous improvement in grid accuracy, especially at resolutions of 256×1024 and 512×2048. Similarly, in Fig. 28, at a resolution of 256×1024, the computational accuracy of WENO5 was comparable to that of TVD at 512×2048, whereas the analytical accuracy of the WENO5 results at a resolution of 512×2048 significantly surpassed the accuracy of TVD at the same grid resolution.

Another noteworthy aspect is the issue of symmetry in the Rayleigh–Taylor instability structure. Although the initial perturbation is generally symmetric, many schemes/solvers could not consistently maintain flow symmetry throughout the computation. Various factors contributed to the appearance of asymmetry. Notably, even for a well-established scheme, such as TVD (which directly uses

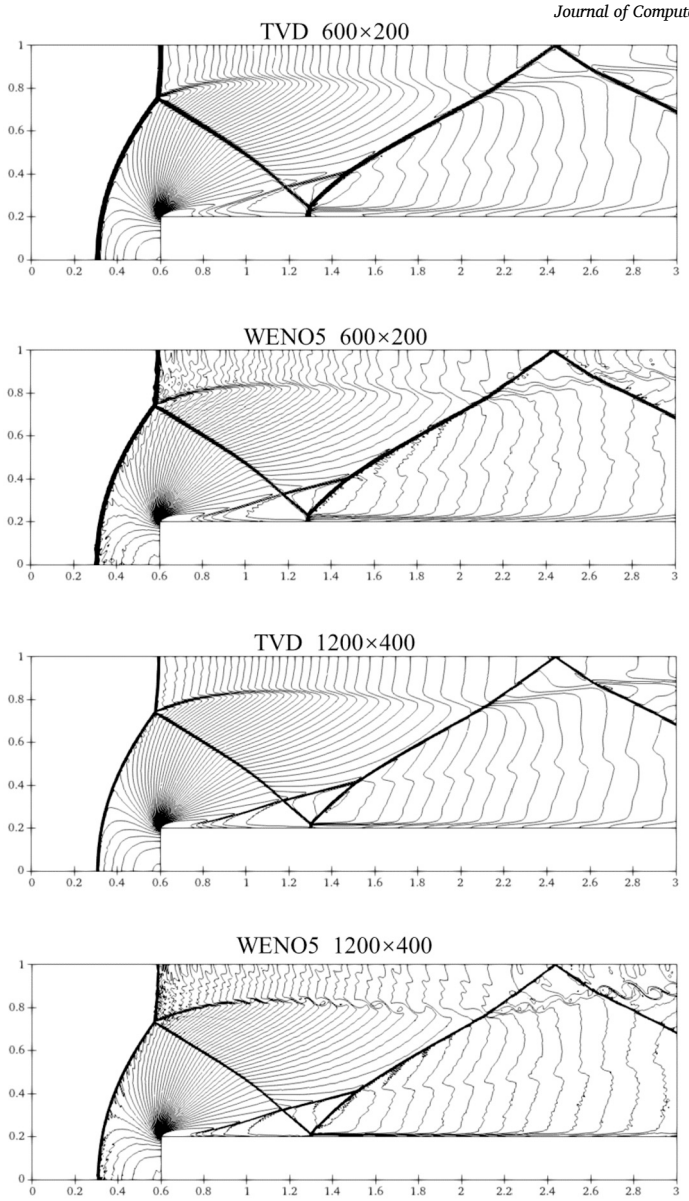

Fig. 25. Density contour line at 4 s for the forward-facing step problem at different schemes and grid resolutions.

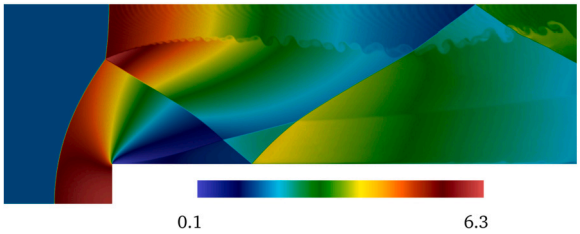

Fig. 26. Density colormap of WENO5-DOLINC using 2400 × 800 resolution. (For interpretation of the colors in the figure(s), the reader is referred to the web version of this article.)

the original code from the OpenFOAM distribution in this study), asymmetry still occurred at a grid resolution of 512×2048, as shown in Figs. 27 and 28. This asymmetry was particularly pronounced in the WENO-P3 scheme, as shown in Fig. 29. The computational results of WENO-P3 exhibited significant asymmetrical features at lower grid resolutions due to the use of multi-grid cell templates and triangular meshing steps on grid faces. In practical computations, the data transfer between processes in parallel computing and certain

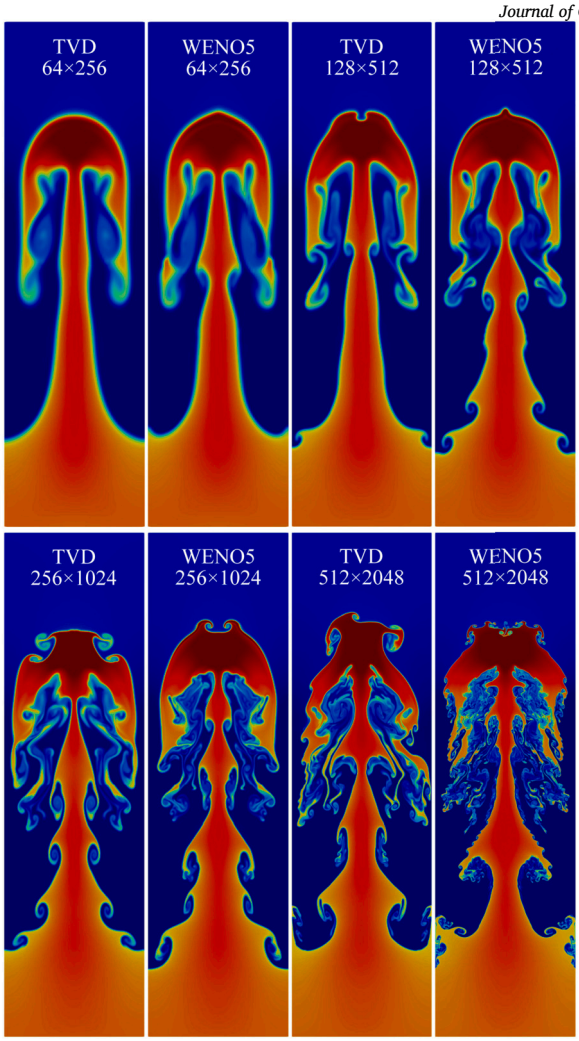

Fig. 27. Density colormap at 1.95 s for the Rayleigh–Taylor instability problem (colormap ranges from 0.85 to 2.25).

treatments using linear-equation iterative solvers for ill-conditioned matrices can also contribute to the emergence of asymmetry. Although the WENO-P3 scheme in Fig. 29 captures more refined flow characteristics, it cannot achieve symmetric results similar to those of high-order structured grid solvers (Fleischmann et al., 2019). However, the WENO5 scheme implemented based on the DOLINC method demonstrated good symmetry, as shown in Figs. 27 and 28, even at resolutions for which the TVD schemes exhibited asymmetric vortex structures. This highlights the advantages of the DOLINC method in implementing high-order unstructured grid schemes. For a more in-depth discussion of the various reasons for asymmetry and methods to improve symmetry, further reading of the relevant literature is recommended [16,36].

In this case, the DOLINC scheme also demonstrated a higher computational efficiency in improving accuracy. The CPU time for the TVD scheme at a grid resolution of 256×1024 using 16 cores was 1089.53 s. Considering this as a benchmark for accuracy and speed, when the grid was refined to 512×2048 without considering the partial effect of parallel acceleration, the original computation time for the TVD scheme was approximately 12712.68 s, increasing the cost by nearly 10.67 times. Even when considering the impact of parallel acceleration, maintaining the same number of grids per computing thread as in the benchmark case and using 64 cores still requires 3878.17 s. When the WENO5-DOLINC scheme was used for computation based on the same grid resolution of 256×1024, then the total time was only 1829.7 s. The cost increased by only 0.68 times compared to the benchmark case and was even faster than the TVD scheme with parallel acceleration at a higher grid resolution. For a high-order unstructured grid scheme implemented using classical k-exact methods and Gauss integration instead of the DOLINC method, the WENO-P3 scheme with similar accuracy at a grid resolution of 256×1024 required 11678.9 s. Thus, the cost increased by nearly 9.72 times compared to the benchmark and was almost equivalent to the TVD scheme with a refined grid of 512×2048 in the low-order FVM architecture. Considering the complexity of implementing such unstructured grid WENO methods in low-order FVM architectures, we can understand why these methods lack appeal for industrial software. At the same grid resolution, the computation time for WENO5 was only 16% that of WENO-P3, reducing the additional computational cost by 93%. Overall, the DOLINC method had a clear advantage over

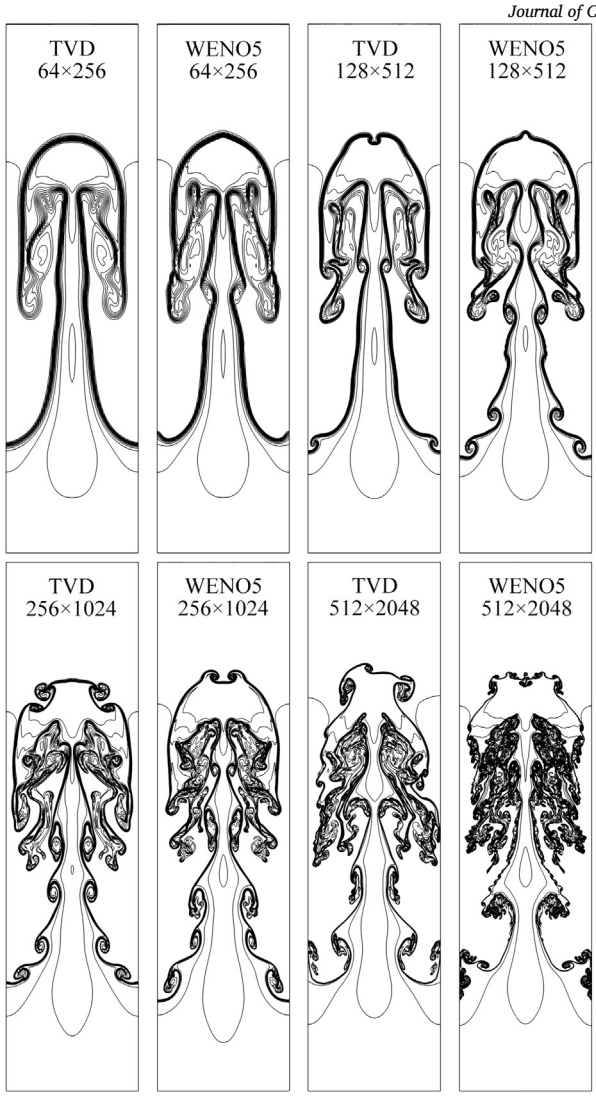

Fig. 28. Density contour lines at 1.95 s for the Rayleigh–Taylor instability problem (contour lines range from 0.95 to 2.15 with 12 lines).

low-order methods in improving computational-accuracy efficiency and outperformed other high-order unstructured grid methods. Further refining the grid to an ultimate resolution of 1024×4096, as shown in Fig. 30, the DOLINC scheme exhibited very fine flow-detail capture and better symmetry, achieving high-precision results similar to those of structured-grid high-order solvers based on low-order FVM architectures [16,39].

3.4.5. Isentropic vortex evolution

The classic isentropic vortex evolution problem was performed to quantitatively verify the actual convergence accuracy of the DOLINC scheme in solving the two-dimensional Euler equations. An isentropic vortex was superimposed on the initial velocity field, with corresponding velocity and temperature perturbations propagating with the mean flow field. In this study, the setup followed the configuration of Shu [40], Hu and Shu [24]. Periodic boundary conditions were employed to study the evolution of the initial superimposed vortex and errors under different schemes. Moreover, grid resolutions were obtained to compute the order of accuracy of the spatial discretization method. However, it is important to note that even after eliminating the error contributed by the time discretization scheme, spatial discretization accuracy remained affected by both the flux scheme and the face interpolation/reconstruction scheme. The former causes the overall spatial convergence order to be lower than the theoretical order of the latter. In this study, we employed three generalized local LF schemes with different accuracies: the basic (termed LF), KT, and KNP types, to perform the calculations [30,29]. Tables 6, 7, and 8 illustrate the errors and convergence rates of the respective schemes.

Tables 6–8 illustrate that owing to errors introduced by the flux scheme, the overall spatial convergence orders of various reconstruction/interpolation schemes did not reach their theoretical orders. For example, regarding the upwind schemes, the first-order theoretical accuracy of interpolation/reconstruction was reduced by flux scheme errors, achieving only a convergence rate of 0.5 in

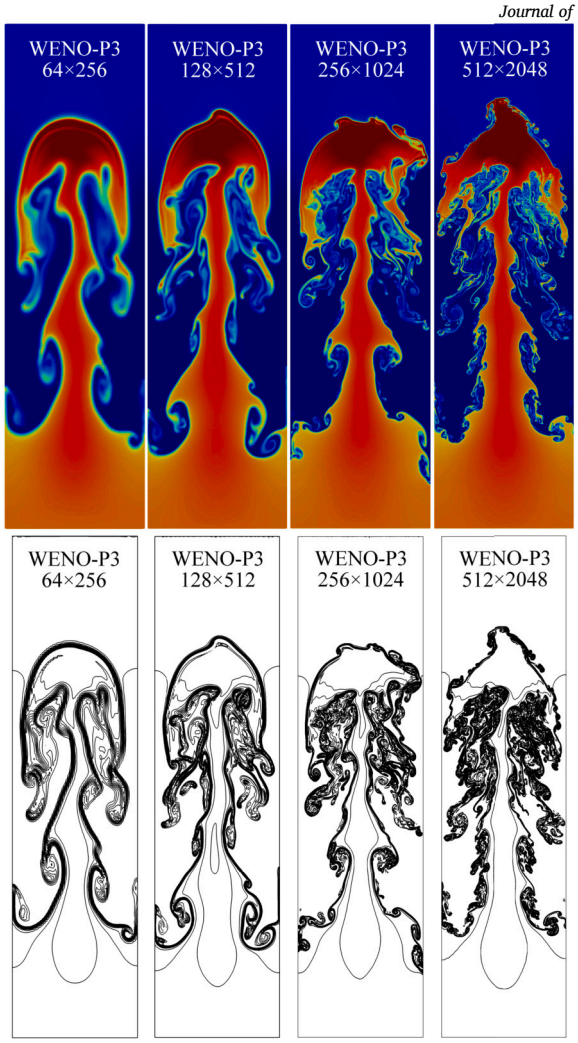

Fig. 29. Colormap and contour lines of density obtained using the WENO-P3 scheme.

Table 6
L_2 error and order of accuracy for different schemes with LF flux scheme.

Mesh	FOU		TVD-VanAlbada		WENO3		WENO5	
	Error	Order	Error	Order	Error	Order	Error	Order
10	1.41E-01		1.19E-01		1.17E-01		9.09E-02	
20	1.37E-01	0.04	8.55E-02	0.48	7.70E-02	0.60	2.48E-02	1.87
40	1.16E-01	0.24	4.19E-02	1.03	2.91E-02	1.40	3.56E-03	2.80
80	8.56E-02	0.43	2.03E-02	1.04	9.92E-03	1.55	2.57E-04	3.80

Table 7
L_2 error and order of accuracy for different schemes with KT flux scheme.

Mesh	FOU		TVD-VanAlbada		WENO3		WENO5	
	Error	Order	Error	Order	Error	Order	Error	Order
10	1.37E-01		1.12E-01		1.09E-01		8.16E-02	
20	1.28E-01	0.10	7.32E-02	0.62	6.47E-02	0.75	1.89E-02	2.11
40	1.02E-01	0.34	3.23E-02	1.18	2.18E-02	1.57	2.55E-03	2.89
80	6.89E-02	0.56	1.47E-02	1.13	7.81E-03	1.48	2.06E-04	3.63

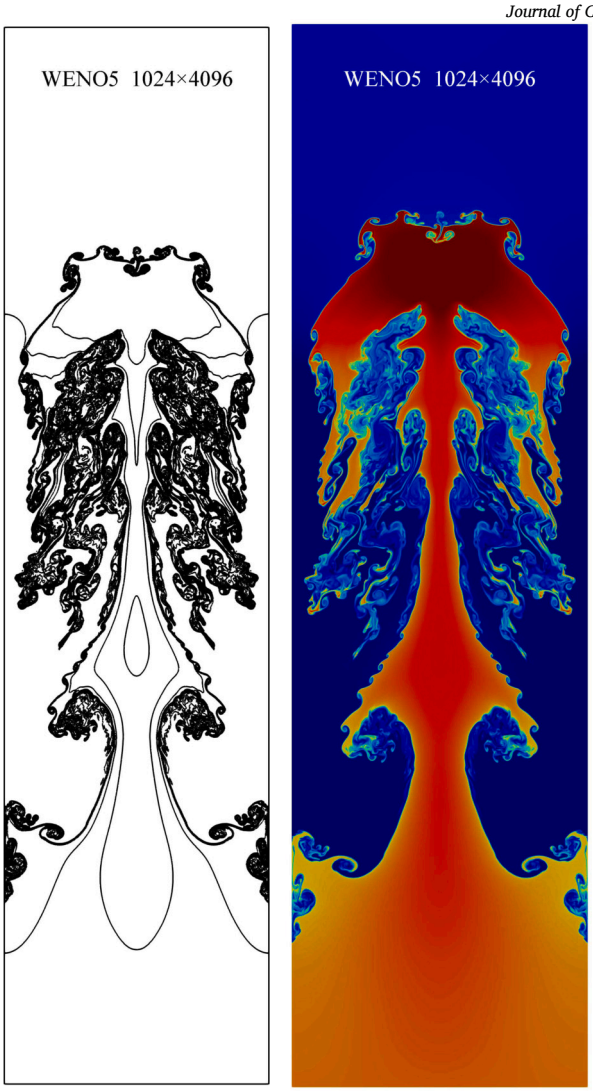

Fig. 30. Contour lines and colormap of density obtained using the WENO5 scheme on the 1024×4096 grid.

Table 8
L_2 error and order of accuracy for different schemes with KNP flux scheme.

Mesh	FOU		TVD-VanAlbada		WENO3		WENO5	
	Error	Order	Error	Order	Error	Order	Error	Order
10	1.22E-01		9.64E-02		8.89E-02		6.06E-02	
20	1.03E-01	0.24	5.19E-02	0.89	4.35E-02	1.03	1.37E-02	2.15
40	7.31E-02	0.50	2.35E-02	1.14	1.49E-02	1.55	1.70E-03	3.01
80	4.52E-02	0.69	1.13E-02	1.06	4.75E-03	1.64	1.72E-04	3.31

practical computations. When more accurate flux schemes were used (progressing from LF to KNP), an improvement in convergence accuracy was observed across all schemes. Therefore, flux scheme errors significantly impacted the overall convergence accuracy, with the flux scheme rather than the interpolation/reconstruction method being the primary limiter on spatial error reductions. Tables 6–8 illustrate the varying effects of different flux schemes on the results, with higher-order schemes generally being more affected by flux scheme errors. This is because the reconstruction errors in higher-order schemes decrease more rapidly, making the flux scheme error a larger component of the overall error. Therefore, the flux scheme significantly influences convergence accuracy.

Fig. 31(a) compares the actual spatial convergence accuracy of various schemes under different flux schemes. It shows that the spatial errors of all four schemes were all influenced by the errors of the chosen flux scheme, with the more accurate one achieving spatial convergence rates closer to the theoretical accuracy of the reconstruction schemes. As shown in Fig. 31(a), the high-order DOLINC schemes suggest an overall advantage over lower-order schemes, with WENO5 schemes achieving a significantly higher

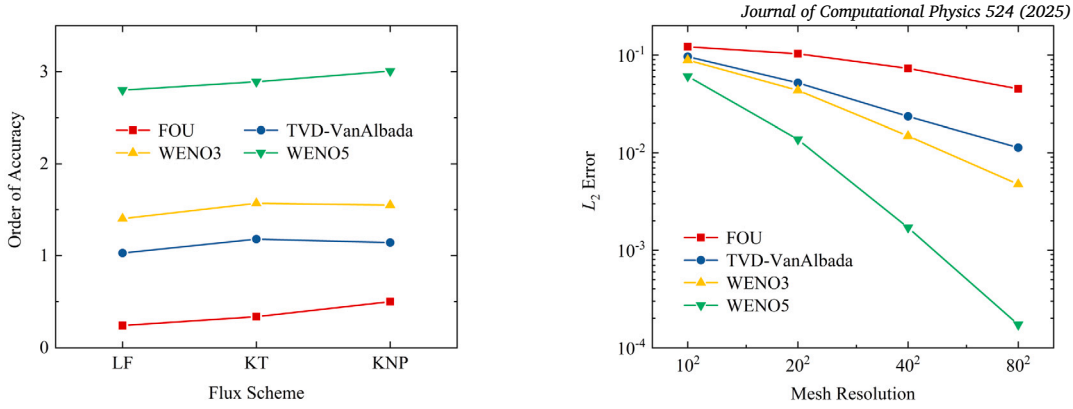

Fig. 31. L_2 errors and convergence orders of different flux-reconstruction-scheme combinations. (*a*) Order of accuracy from grid number 20^2 to 40^2, (*b*) errors of different reconstruction schemes with KNP.

convergence rate than others. This study focused on the face interpolation schemes implemented by the DOLINC method. A detailed analysis of flux scheme or Riemann solver influence is beyond the scope of this study; thus, no further discussion on flux schemes proceeds.

Despite being influenced by the flux scheme, the advantage of the DOLINC high-order schemes in improving spatial discretization accuracy remains clear. Compared to the first-order upwind and TVD schemes, the WENO3 and WENO5 schemes exhibited significantly faster convergence rates, as shown in Fig. 31(*b*). When WENO5 was combined with the KNP method, it achieved a convergence rate over the third order (Table 8). Moreover, the error of the WENO5-DOLINC scheme decreased much faster as grid resolution is high, which exhibited the non-linear behavior in double logarithmic Fig. 31(*b*) as a consequence of flux-scheme influence.

Fig. 32 shows the pressure distribution at $x = 0$ after 5 and 10 periods for different schemes. The mesh resolution employed is 80^2. It is clearly observable from the figure that the original TVD scheme exhibited strong dissipative characteristics, almost flattening the pressure extrema after 10 periods. In comparison, the two WENO schemes implemented by the DOLINC method demonstrated significantly better accuracy, particularly WENO5, which exhibited only minor pressure dissipation after 10 periods. Moreover, Fig. 32(*b*) presents a modified TVD-V scheme that applies a special treatment to the vector field. Although the dissipation of the modified TVD-V scheme was still slightly stronger than that of WENO3, it showed a marked improvement over the primitive TVD scheme. It is also noted that both WENO schemes achieved relatively symmetric results in a diagonal flow environment, whereas the TVD schemes displayed noticeable asymmetry after 10 periods, regardless of vector treatment.

The two-dimensional distributions after 4 periods in Figs. 33 and 34 provide a more intuitive reflection of the characteristics discussed above. Fig. 33 displays the density distribution results for different schemes, where the strong dissipation of the original TVD scheme, the improvements of the TVD-V scheme, and the high accuracy of WENO5 are all distinctly visible. No significant asymmetry can be observed in the density distribution alone. However, this asymmetry becomes clear in the vorticity magnitude shown in Fig. 34. The vorticity of TVD scheme demonstrated strong dissipation and anisotropy, whereas WENO5, in contrast, essentially maintained the initial isotropic distribution. Compared to WENO5, WENO3 displayed slight anisotropic stretching along the diagonal flow direction. Although the dissipation characteristics of the TVD-V and WENO3 schemes were similar, the former exhibited stronger diffusion along the flow direction. These results highlight the excellent symmetry-preserving properties of the higher-order DOLINC schemes in diagonal flow conditions.

3.4.6. Scramjet inlet simulation

We test the computational performance of the DOLINC method on more irregular grids and demonstrate its applicability to engineering problems by using the WENO5-DOLINC scheme to compute the flow in the inlet of a more complex scramjet engine. Two different forms of computational domains were utilized to simulate different engine conditions, as shown in Fig. 35. Model A was used for the scramjet engine operating under the design conditions with a Mach 8.5 inflow. Three converging compression shocks occurred at the lower edge, point 3. In this case, the polyhedral mesh exhibited a certain degree of skewness and noticeable non-uniformity. Model B was employed for the non-design conditions of the scramjet engine, typically involving a lower cruise Mach number of an aircraft. In this case, the incident position of the converging oblique shock moved forward, striking the lower boundary of the computational domain and requiring the use of non-reflective boundary conditions. The non-uniformity of the mesh in the inlet section of Model B was further enhanced.

The Mach number contours obtained using the WENO5-DOLINC scheme for the design condition with an incoming Mach number of 8.5 and non-design condition with an incoming Mach number of 5.0 are shown in Fig. 36. The scales of the horizontal and vertical axes were adjusted to facilitate the display of the morphological structure of the shock/expansion wave systems. For the design condition, the various wave structures in both Models A and B were consistent, confirming that the designed scramjet-engine geometry achieved the intended goal of correctly converging three oblique shocks at the lower-wall front point at the Mach 8.5 inflow. In the results for the Mach 5.0 inflow in Fig. 36(*c*), the non-reflective bottom boundary smoothly passed through the oblique compression shock

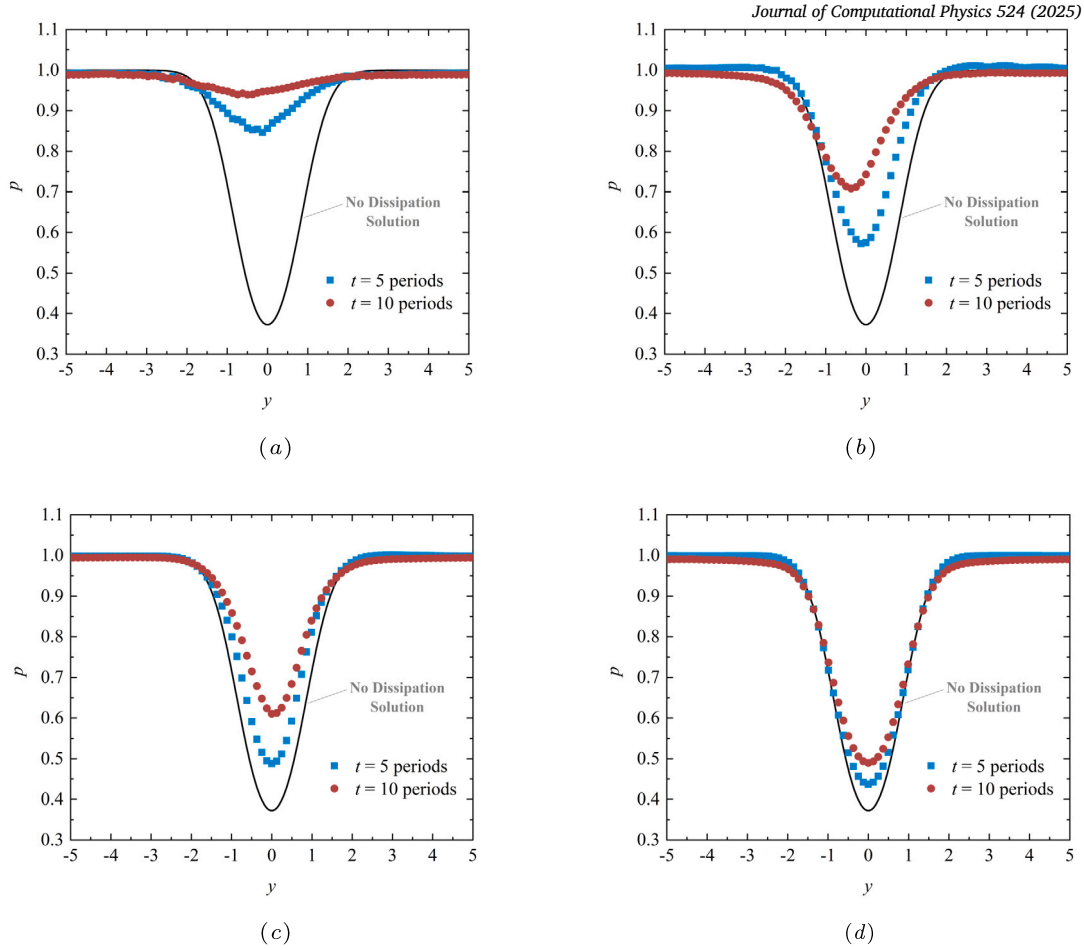

Fig. 32. Pressure distribution of $x = 0$ after several periods of transport. (*a*) TVD scheme, (*b*) TVD-V scheme, (*c*) WENO3 scheme, (*d*) WENO5 scheme.

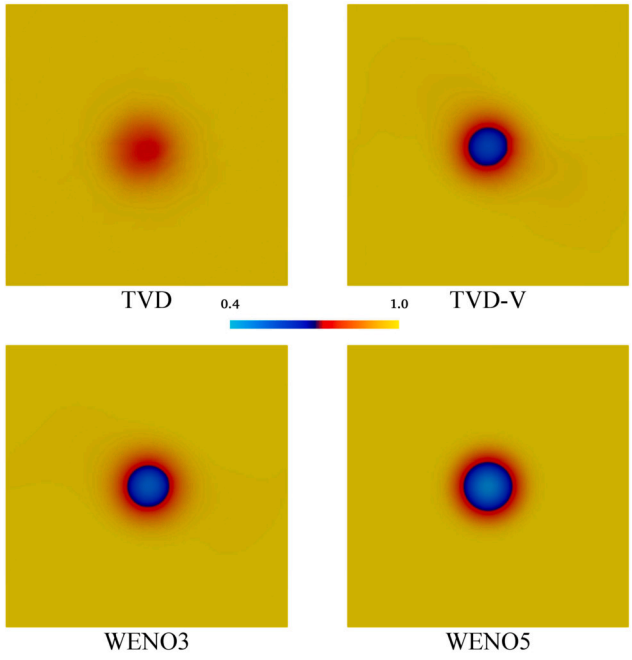

Fig. 33. Density of the isentropic vortex evolution problem at 40 s.

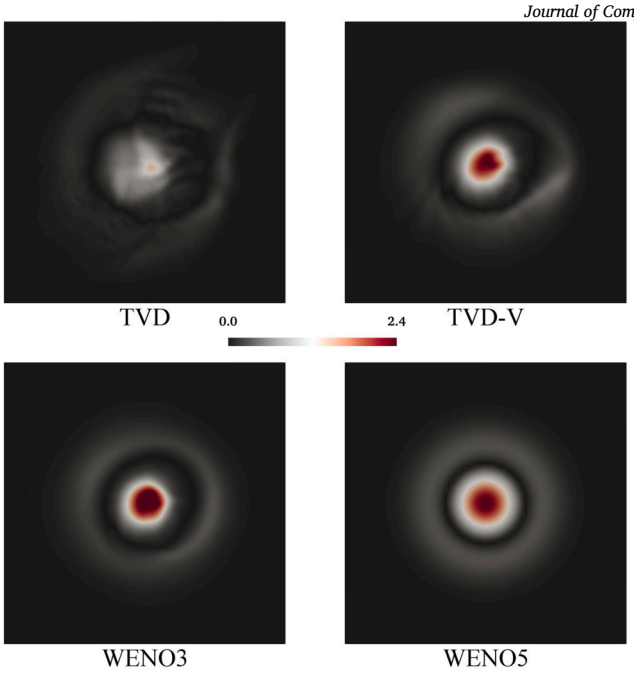

Fig. 34. Vorticity magnitude of the isentropic vortex evolution problem at 40 s.

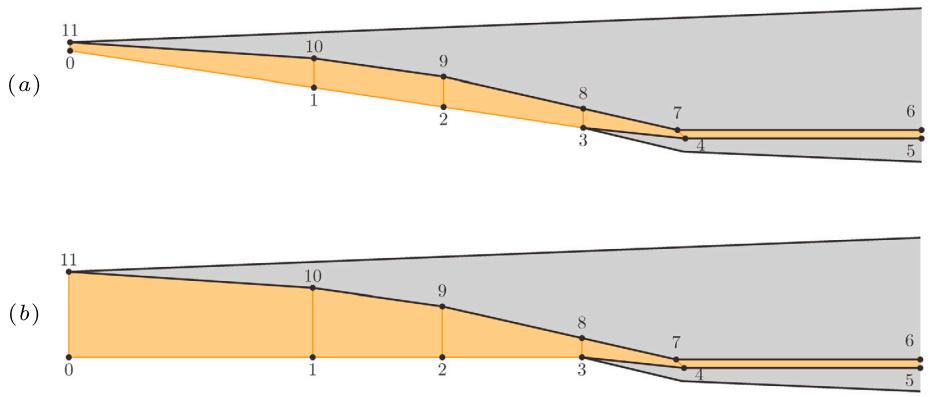

Fig. 35. Schematic of the inlet geometry and computational domain. (*a*) Domain model A for the design condition, (*b*) domain model B for the generic condition.

without forming any reflected shock waves to interfere with the downstream computational domain. Overall, even in grids with significant non-uniformity and some skewness, the DOLINC method could still obtain high-precision results. The DOLINC scheme demonstrated high accuracy in capturing multiple complex reflection processes and the interaction between shock/expansion waves in the isolator section without unexpected dissipation or oscillations owing to the large grid-scale differences between the inlet and isolator sections.

The outlet parameters of the isolator section are crucial in the system modeling and simulation process of scramjet engines because they are related to the design targets of the downstream combustion chambers. Therefore, a dedicated study was conducted on the variations in the isolator section parameters along the flow path, as shown in Figs. 37 and 38. Fig. 37 presents the along-path variations in the static temperature/pressure on the upper wall, lower wall, and cross-sectional average in the isolator section under the design conditions. Although the isolator section experienced continuous shock reflections and interactions between shock and expansion waves, the average pressure and temperature remained relatively stable. In Fig. 38, under non-design conditions, the front end of the isolator section may exhibit more complex wave structures and parameter fluctuations compared with the design condition; however, the variations in the averages remain stable within a certain range. From the along-path distributions, we can observe that at the Mach 8.5 inflow condition, the pressure-fluctuation range at the isolator section outlet was approximately 15 kPa, and the temperature-fluctuation range was approximately 130 K. When the inflow Mach number was reduced to 5.0, the pressure-fluctuation range decreased to approximately 8 kPa, and the temperature-fluctuation range decreased to approximately 15 K. The design of the downstream combustion chambers should also consider the possible deviations in these corresponding parameters. Therefore, accurate prediction of the flow in the engine inlet using the DOLINC method provides more reliable practical references for simplifying

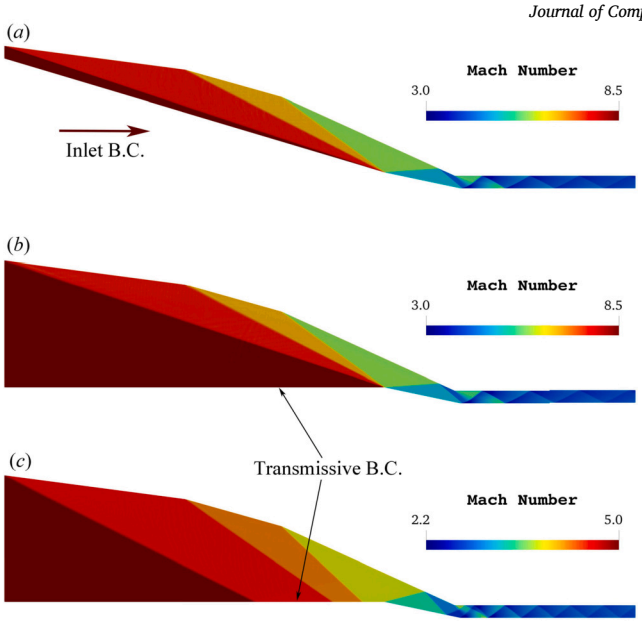

Fig. 36. Mach number of the scramjet inlet under different operating conditions. (*a*) Domain model A with inflow Mach number of 8.5, (*b*) domain model B with inflow Mach number of 8.5, and (*c*) domain model B with inflow Mach number of 5.0.

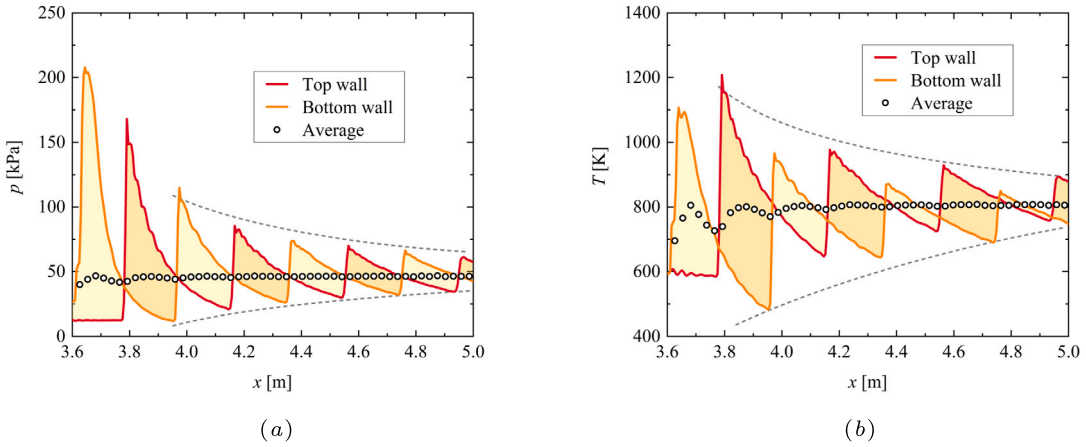

Fig. 37. Parameter distribution along isolator path under Mach 8.5 inflow conditions. (*a*) Static pressure, (*b*) static temperature.

assumptions in system-level simulations (such as which incoming shocks/expansion waves can be ignored or simplified and which must be precisely modeled) and offers more accurate boundary parameters for other component-level design work, thereby improving the overall design effectiveness.

4. Conclusions

The present study introduces a face-neighboring cell order-lifted inversion approach, the DOLINC method, to achieve high-order schemes within a low-order FVM framework based on unstructured polyhedral grids. This method successfully implemented various high-order schemes, including the fixed-template reconstruction and ENO/WENO methods, in a multi-dimensional second-order FVM framework. This study evaluated the computational accuracy and efficiency of the DOLINC method using different computational cases, leading to the following main conclusions:

(1) In principle, the DOLINC method is applicable to any polyhedral grid stored in an unstructured data format. By increasing the order, the DOFs required for high-order schemes were converted into DOLINC differentials. The inversion formulas derived in this study enabled the precise computation of the far-field cell data required for high-order schemes on non-uniform hexahedral grids.

(2) The results from one-dimensional and two-dimensional cases indicated that using the DOLINC scheme in unstructured grid solvers could achieve analytical accuracy comparable to that of high-order solvers on structured grids, as reported in existing studies.

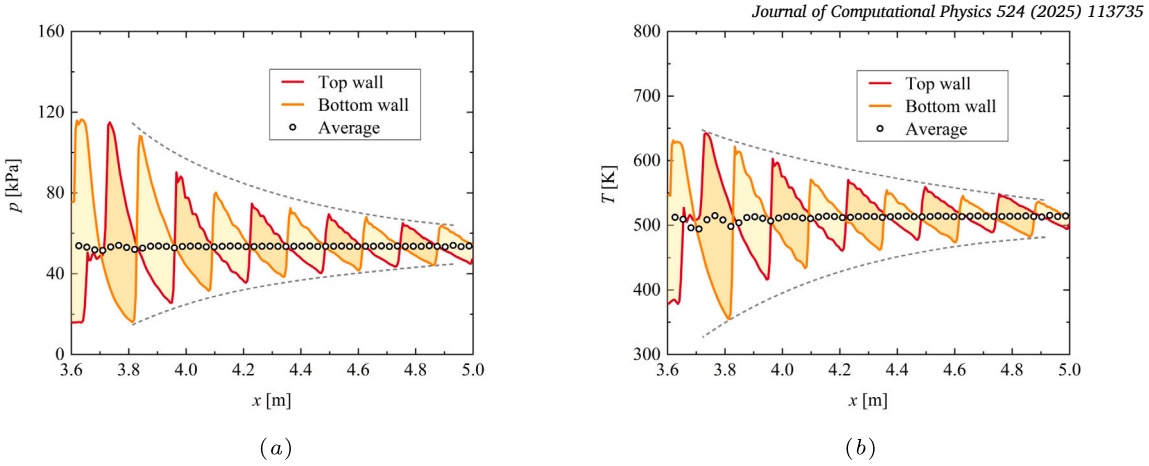

Fig. 38. Parameter distribution along isolator path under Mach 5.0 inflow conditions. (*a*) Static pressure, (*b*) static temperature.

(3) In terms of efficiently improving the computational accuracy, the DOLINC method had an advantage over low-order FVM using refined grids. Compared to the baseline schemes, CD2/TVD, on fine grids, CSR4-DOLINC and WENO5-DOLINC achieved faster computation on coarse grids with higher accuracy, reducing the CPU time by approximately 50% and 70%, respectively.

(4) When implementing similar high-order schemes with comparable accuracy on unstructured grids, the DOLINC method exhibited a noticeable speed advantage over the classical k-exact method. In addition, it demonstrated better symmetry in regular grid problems.

(5) For meshes characterized by significant variations in local cell size, such as boundary-layer meshes, the accuracy of the DOLINC method was significantly higher than that of the uniform-mesh-based least-squares gradient approximation.

(6) The DOLINC method performed well for a wide range of problems, including pressure-coupling algorithms, Riemann solvers, and passive scalar transport. This effectively balanced accuracy and computational costs. The proposed method was characterized by simple implementation, did not require special boundary handling, and exhibited high parallel efficiency. It could be easily applied to the existing low-order FVM frameworks adopted by most industrial CFD software programs with minimal modifications.

To summarize, the DOLINC method combined the advantages of easy implementation, high computational speed, and high solution accuracy. This contributed to the current trend of industrial software transitioning to higher-order methods, thereby facilitating efficient and accurate solutions to complex industrial problems. The main limitation of the DOLINC method currently lies in its low-accuracy inversion formula for tetrahedral or hybrid grids, which potentially leads to accuracy degradation. However, this issue can be addressed by developing specific inversion formulas for particular types of elements, indicating a direction for future improvements to the DOLINC method.

CRediT authorship contribution statement

Hao Guo: Writing – review & editing, Writing – original draft, Visualization, Validation, Software, Methodology, Investigation, Formal analysis, Conceptualization. **Boxing Hu:** Writing – original draft, Methodology, Conceptualization. **Peixue Jiang:** Supervision, Resources, Investigation. **Xiaofeng Ma:** Writing – original draft, Validation, Methodology, Data curation, Conceptualization. **Yinhai Zhu:** Writing – review & editing, Validation, Supervision, Resources, Project administration, Methodology, Investigation, Conceptualization.

Code availability

The source code used to reproduce the results of this study will be openly available on GitHub at FVM-High-Older-DOLINC-Scheme upon publication.

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 study was supported by the National Natural Science Foundation of China (U23B20108) and the National Science and Technology Major Project (Grant No. J2019-III-0019-0063).

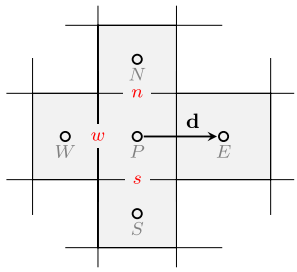

Fig. A.39. Schematic of two-dimensional mesh with distance vector **d** pointing from cell P to E.

Appendix A. Derivation details of the DOLINC differential and inversion

Without loss of generality, a two-dimensional non-uniform mesh presented in Fig. A.39 was employed to demonstrate the inversion of far-field cell values. As addressed in Section 2.3, the linear interpolation together with Gauss's theorem should be used for gradient calculations:

$$\bar{\phi}_{P,j} = \overline{\left(\frac{\partial \phi}{\partial x_j}\right)}_P = \frac{\phi_w S_{w,j} + \phi_e S_{e,j} + \phi_n S_{n,j} + \phi_s S_{s,j}}{V}. \tag{A.1}$$

Here, ϕ can be any-order tensor and $S_{w,j}$ represents the jth component of western face vector $\mathbf{S}_w$. If we align x coordinate direction with west to east and focus on the x component of gradient only, because of the elimination of $S_{n,x}$ and $S_{s,x}$, the equation becomes

$$\bar{\phi}_{P,x} = \frac{\phi_w S_{w,x} + \phi_e S_{e,x}}{V}. \tag{A.2}$$

This is identical to a one-dimensional formula, while ϕ and coordinate vectors still being multi-dimensional, called x-split equation of a multi-dimensional domain. Based on the assumptions of non-uniform mesh in Section 2.2, we obtain

$$\phi_w = \omega \bar{\phi}_W + (1-\omega)\bar{\phi}_P, \tag{A.3}$$

$$\phi_e = \omega \bar{\phi}_P + (1-\omega)\bar{\phi}_E. \tag{A.4}$$

Substituting Equations (A.3) and (A.4) into Equation (A.2) and rearranging, we obtain

$$-\omega\bar{\phi}_W + (2\omega - 1)\bar{\phi}_P + (1-\omega)\bar{\phi}_E = \bar{\phi}_{P,x}\, V / S_{,(x)}. \tag{A.5}$$

We denote cell width in the x direction as Δ. Using Equations (2) and (4), the distance vector magnitude d can be expressed as

$$d = \frac{1}{2}\Delta(1+g) = \frac{\Delta}{2(1-\omega)}. \tag{A.6}$$

Employing inner product $\bar{\phi}_{P,j} d_j$ for the multi-dimensional space vector (skewness can be handled by conventional FVM strategies), the relation now becomes

$$-\omega\bar{\phi}_W + (2\omega - 1)\bar{\phi}_P + (1-\omega)\bar{\phi}_E = 2(1-\omega)\bar{\phi}_{P,j} d_j. \tag{A.7}$$

Dividing by $1-\omega$ on both sides and using Equation (4), we obtain

$$-g\bar{\phi}_W + (g-1)\bar{\phi}_P + \bar{\phi}_E = 2\bar{\phi}_{P,j} d_j. \tag{A.8}$$

Rearranged with subscripts W, P and E changed to $i-1$, i and $i+1$, we have

$$\bar{\phi}_{i+1} = g\bar{\phi}_{i-1} + 2d_j\bar{\phi}_{i,j} + (1-g)\bar{\phi}_i. \tag{A.9}$$

Similarly, the further west stencil can be adopted to compute values of distant grid cells while distance vector magnitude d increases with cell size:

$$\bar{\phi}_{i+n+2} = g\bar{\phi}_{i+n} + 2(g^{n+1} d_j)\bar{\phi}_{i+n+1,j} + (1-g)\bar{\phi}_{i+n+1}. \tag{A.10}$$

Here, distance vector **d** still points from cell center i to $i+1$. This is exactly the inversion equation (8) in Section 2.3. Note that Equation (A.10) is accurate and error-free as long as the gradient calculation method in Equation (A.1) strictly employs simple linear interpolation together with the Gauss theorem. Further, no assumption was made for tensor order of physical quantity $\bar{\phi}$. It can also be the calculated gradient $\bar{\phi}_{P,j}$. With the same procedure, we can achieve

$$\bar{\phi}_{i+n+2,j} = g\bar{\phi}_{i+n,j} + 2(g^{n+1} d_k)\bar{\phi}_{i+n+1,jk} + (1-g)\bar{\phi}_{i+n+1,j}, \tag{A.11}$$

where higher-order DOLINC differentials occur and summation only applies for subscript k. Contracting both sides of the equation with $2d_j$ yields the first-order DOLINC differential as

$$2d_j\bar{\phi}_{i+n+2,j} = g\left[2d_j\bar{\phi}_{i+n,j} + g^n(2d_jd_k\bar{\phi}_{i+n+1,jk})\right] + (1-g)(2d_j\bar{\phi}_{i+n+1,j}). \tag{A.12}$$

Similarly for second-order DOLINC differential we have

$$4d_jd_k\bar{\phi}_{i+n+2,jk} = g\left[4d_jd_k\bar{\phi}_{i+n,jk} + g^n(4d_jd_kd_t\bar{\phi}_{i+n+1,jkt})\right] + (1-g)(4d_jd_k\bar{\phi}_{i+n+1,jk}). \tag{A.13}$$

Generally, Equation (13) can be derived for an m-th order DOLINC differential with some notations defined in Section 2.3.

Data availability

Data will be made available on request.

References

[1] R. Abgrall, On essentially non-oscillatory schemes on unstructured meshes: Analysis and implementation, J. Comput. Phys. 114 (1994) 45–58, https://doi.org/10.1006/jcph.1994.1148.
[2] Ansys Fluent, Fluent user guide, 2016.
[3] F. Archambeau, N. Méchitoua, M. Sakiz, Code_saturne: a finite volume code for the computation of turbulent incompressible flows, Int. J. Finite Vol. (2004).
[4] R. Aris, Vectors, tensors, and the basic equations of fluid mechanics, Prentice-Hall, Inc., 1962.
[5] D.S. Balsara, M. Dumbser, R. Abgrall, Multidimensional hllc riemann solver for unstructured meshes – with application to euler and mhd flows, J. Comput. Phys. 261 (2014) 172–208, https://doi.org/10.1016/j.jcp.2013.12.029.
[6] T. Barth, P. Frederickson, Higher order solution of the euler equations on unstructured grids using quadratic reconstruction, in: 28th Aerospace Sciences Meeting, 1990.
[7] M. Brio, A. Zakharian, G. Webb, Two-dimensional Riemann solver for Euler equations of gas dynamics, J. Comput. Phys. 167 (2001) 177–195, https://doi.org/10.1006/jcph.2000.6666.
[8] X. Chai, P.S. Iyer, K. Mahesh, Numerical study of high speed jets in crossflow, J. Fluid Mech. 785 (2015) 152–188, https://doi.org/10.1017/jfm.2015.612.
[9] M. Darwish, F. Moukalled, Tvd schemes for unstructured grids, Int. J. Heat Mass Transf. 46 (2003) 599–611, https://doi.org/10.1016/s0017-9310(02)00330-7.
[10] 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, https://doi.org/10.1016/j.jcp.2006.06.043.
[11] M. Dumbser, M. Käser, V.A. Titarev, E.F. Toro, Quadrature-free non-oscillatory finite volume schemes on unstructured meshes for nonlinear hyperbolic systems, J. Comput. Phys. 226 (2007) 204–243, https://doi.org/10.1016/j.jcp.2007.04.004.
[12] L.J. Durlofsky, B. Engquist, S. Osher, Triangle based adaptive stencils for the solution of hyperbolic conservation laws, J. Comput. Phys. 98 (1992) 64–73, https://doi.org/10.1016/0021-9991(92)90173-v.
[13] J.A. Ekaterinaris, High-order accurate, low numerical diffusion methods for aerodynamics, Prog. Aerosp. Sci. 41 (2005) 192–300, https://doi.org/10.1016/j.paerosci.2005.03.003.
[14] ESI-OpenCFD, OpenFOAM user guide, 2021.
[15] P. Fan, Y. Shen, B. Tian, C. Yang, A new smoothness indicator for improving the weighted essentially non-oscillatory scheme, J. Comput. Phys. 269 (2014) 329–354, https://doi.org/10.1016/j.jcp.2014.03.032.
[16] N. Fleischmann, S. Adami, N.A. Adams, Numerical symmetry-preserving techniques for low-dissipation shock-capturing schemes, Comput. Fluids 189 (2019) 94–107, https://doi.org/10.1016/j.compfluid.2019.04.004.
[17] O. Friedrich, Weighted essentially non-oscillatory schemes for the interpolation of mean values on unstructured grids, J. Comput. Phys. 144 (1998) 194–212, https://doi.org/10.1006/jcph.1998.5988.
[18] H. Guo, P. Jiang, W. Peng, Y. Zhu, Large eddy simulation of highly compressible film cooling in transonic crossflow, Int. J. Heat Mass Transf. 202 (2023) 123765, https://doi.org/10.1016/j.ijheatmasstransfer.2022.123765.
[19] H. Guo, P. Jiang, L. Ye, Y. Zhu, An efficient and low-divergence method for generating inhomogeneous and anisotropic turbulence with arbitrary spectra, J. Fluid Mech. 970 (2023) A2, https://doi.org/10.1017/jfm.2023.548.
[20] H. Guo, P. Jiang, Y. Zhu, Development of low-dissipative projection method framework integrating various high-order time integration schemes using openfoam, 2023, arXiv.
[21] J.W. Gärtner, A. Kronenburg, T. Martin, Efficient weno library for openfoam, SoftwareX 12 (2020) 100611, https://doi.org/10.1016/j.softx.2020.100611.
[22] A. Harten, High resolution schemes for hyperbolic conservation laws, J. Comput. Phys. 135 (1997) 260–278, https://doi.org/10.1006/jcph.1997.5713.
[23] A. Harten, B. Engquist, S. Osher, S.R. Chakravarthy, Uniformly high order accurate essentially non-oscillatory schemes. III, J. Comput. Phys. 71 (1987) 231–303, https://doi.org/10.1016/0021-9991(87)90031-3.
[24] C. Hu, C.W. Shu, Weighted essentially non-oscillatory schemes on triangular meshes, J. Comput. Phys. 150 (1999) 97–127, https://doi.org/10.1006/jcph.1998.6165.
[25] H.T. Huynh, A flux reconstruction approach to high-order schemes including discontinuous Galerkin methods, in: 18th AIAA Computational Fluid Dynamics Conference, 2007.
[26] H.T. Huynh, A reconstruction approach to high-order schemes including discontinuous Galerkin for diffusion, in: 47th AIAA Aerospace Sciences Meeting including The New Horizons Forum and Aerospace Exposition, 2009.
[27] H.T. Huynh, High-order methods by correction procedures using reconstructions, World Sci. (2011) 391–422, https://doi.org/10.1142/9789814313193_0014.
[28] G.S. Jiang, C.W. Shu, Efficient implementation of weighted ENO schemes, J. Comput. Phys. 126 (1996) 202–228, https://doi.org/10.1006/jcph.1996.0130.
[29] A. Kurganov, S. Noelle, G. Petrova, Semidiscrete central-upwind schemes for hyperbolic conservation laws and Hamilton–Jacobi equations, SIAM J. Sci. Comput. 23 (2001) 707–740, https://doi.org/10.1137/s1064827500373413.
[30] A. Kurganov, E. Tadmor, New high-resolution central schemes for nonlinear conservation laws and convection-diffusion equations, J. Comput. Phys. 160 (2000) 241–282, https://doi.org/10.1006/jcph.2000.6459.
[31] P.D. Lax, Weak solutions of nonlinear hyperbolic equations and their numerical computation, Commun. Pure Appl. Math. 7 (1954) 159–193, https://doi.org/10.1002/cpa.3160070112.
[32] W. Li, Y.X. Ren, High-order k-exact weno finite volume schemes for solving gas dynamic euler equations on unstructured grids, Int. J. Numer. Methods Fluids 70 (2011) 742–763, https://doi.org/10.1002/fld.2710.
[33] X.D. Liu, S. Osher, T. Chan, Weighted essentially non-oscillatory schemes, J. Comput. Phys. 115 (1994) 200–212, https://doi.org/10.1006/jcph.1994.1187.

[34] T. Martin, I. Shevchuk, Implementation and validation of semi-implicit WENO schemes using OpenFOAM®, Computation 6 (2018) 6, https://doi.org/10.3390/computation6010006.
[35] C.F. Ollivier-Gooch, Quasi-ENO schemes for unstructured meshes based on unlimited data-dependent least-squares reconstruction, J. Comput. Phys. 133 (1997) 6–17, https://doi.org/10.1006/jcph.1996.5584.
[36] J.F. Remacle, J.E. Flaherty, M.S. Shephard, An adaptive discontinuous galerkin technique with an orthogonal basis applied to compressible flow problems, SIAM Rev. 45 (2003) 53–72, https://doi.org/10.1137/s00361445023830.
[37] C.W. Schulz-Rinne, J.P. Collins, H.M. Glaz, Numerical solution of the Riemann problem for two-dimensional gas dynamics, SIAM J. Sci. Comput. 14 (1993) 1394–1414, https://doi.org/10.1137/0914082.
[38] C. Sheng, Q. Zhao, D. Zhong, N. Ge, A strategy to implement high-order WENO schemes on unstructured grids, in: AIAA Aviation 2019 Forum, American Institute of Aeronautics and Astronautics, 2019.
[39] J. Shi, Y.T. Zhang, C.W. Shu, Resolution of high order weno schemes for complicated flow structures, J. Comput. Phys. 186 (2003) 690–696, https://doi.org/10.1016/S0021-9991(03)00094-9.
[40] C.W. Shu, Essentially non-oscillatory and weighted essentially non-oscillatory schemes for hyperbolic conservation laws, Springer, Berlin, Heidelberg, 1998, pp. 325–432.
[41] Siemens, STAR-CCM+ user guide, 2015.
[42] G.A. Sod, A survey of several finite difference methods for systems of nonlinear hyperbolic conservation laws, J. Comput. Phys. 27 (1978) 1–31, https://doi.org/10.1016/0021-9991(78)90023-2.
[43] M. Sun, Y. Liu, Z. Hu, Turbulence decay in a supersonic boundary layer subjected to a transverse sonic jet, J. Fluid Mech. 867 (2019) 216–249, https://doi.org/10.1017/jfm.2019.158.
[44] E.F. Toro, Riemann Solvers and Numerical Methods for Fluid Dynamics, Springer, 1997.
[45] P. Tsoutsanis, A.F. Antoniadis, D. Drikakis, WENO schemes on arbitrary unstructured meshes for laminar, transitional and turbulent flows, J. Comput. Phys. 256 (2014) 254–276, https://doi.org/10.1016/j.jcp.2013.09.002.
[46] 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, https://doi.org/10.1016/j.jcp.2010.11.023.
[47] C. Wagner, T. Hüttl, P. Sagaut, Large-Eddy Simulation for Acoustics, Cambridge University Press, 2007.
[48] Z. Wang, L. Shi, S. Fu, H. Zhang, L. Zhang, A $P_N P_M$-CPR framework for hyperbolic conservation laws, in: 20th AIAA Computational Fluid Dynamics Conference, 2011.
[49] Z.J. Wang, High-order methods for the euler and Navier–Stokes equations on unstructured grids, Prog. Aerosp. Sci. 43 (2007) 1–41, https://doi.org/10.1016/j.paerosci.2007.05.001.
[50] Z.J. Wang, H. Gao, A unifying lifting collocation penalty formulation including the discontinuous Galerkin, spectral volume/difference methods for conservation laws on mixed grids, J. Comput. Phys. 228 (2009) 8161–8186, https://doi.org/10.1016/j.jcp.2009.07.036.
[51] P. Woodward, P. Colella, The numerical simulation of two-dimensional fluid flow with strong shocks, J. Comput. Phys. 54 (1984) 115–173, https://doi.org/10.1016/0021-9991(84)90142-6.
[52] D. Zhong, C. Sheng, A new method towards high-order WENO schemes on structured and unstructured grids, Comput. Fluids 200 (2020) 104453, https://doi.org/10.1016/j.compfluid.2020.104453.