

A refined r -factor algorithm for TVD schemes on arbitrary unstructured meshes

Di Zhang^{1,*}, Chunbo Jiang^{1,*}, Liang Cheng^{2,3}, Dongfang Liang⁴

¹*State Key Laboratory of Hydrosience and Engineering, Tsinghua University, Beijing 100084, China*

²*School of Civil, Environmental and Mining Engineering, The University of Western Australia, 35 Stirling Highway, Crawley, WA 6009, Australia*

³*State Key Laboratory of Coastal and Offshore Engineering, Dalian University of Technology, Dalian, 116024, China*

⁴*Department of Engineering, University of Cambridge, Trumpington Street, Cambridge CB2 1PZ, UK*

SUMMARY

A refined r -factor algorithm for implementing TVD schemes on arbitrary unstructured meshes, referred to henceforth as FFISAM (a Face-perpendicular Far-upwind Interpolation Scheme for Arbitrary Meshes), is proposed based on an extensive review of the existing r -factor algorithms available in the literature. The design principles, as well as the respective advantages and disadvantages, of the existing algorithms are first systematically analyzed before presenting the FFISAM. The FFISAM is designed to combine the merits of various existing r -factor algorithms. The performance of the FFISAM, implemented in ten classical TVD schemes, is evaluated against four two-dimensional pure-advection benchmark test cases where analytical solutions are available. The numerical results clearly show that the FFISAM leads to a better overall performance than the existing algorithms in terms of accuracy and convergence on arbitrary unstructured meshes for the ten classical TVD schemes.

Keywords: total variation diminishing (TVD); high resolution scheme (HRS); normalized variable diagram (NVD); r -factor algorithm; convection discretization; unstructured meshes

* Corresponding authors: State Key Laboratory of Hydrosience and Engineering, Tsinghua University, The Hydraulic Building, Beijing 100084, China. Phone: +86 010 6278 1820.

E-mail address: zhangdi10@mails.tsinghua.edu.cn (D. Zhang), jcb@mail.tsinghua.edu.cn (C. Jiang).

1. Introduction

Numerical simulation of convection-dominated flows remains one of the most challenging problems in computational fluid dynamics [1-4]. It is well known that conventional low-order (LO) schemes, such as the first-order upwind (FOU), Hybrid and Power-Law schemes, although highly stable and unconditionally bounded, suffer from excessive numerical diffusion [5–7]. However, traditional high-order (HO) schemes, such as the central differencing (CD), second-order upwind (SOU), cubic-upwind interpolation (CUI) and quadratic-upwind interpolation (QUICK) schemes, although being able to improve the accuracy in smooth regions, often generate unphysical oscillations in discontinuous or steep-gradient regions because of their unbounded nature [8-10].

The endowment of the boundedness property to the HO schemes results in the so-called high-resolution schemes (HRS), which are able to provide good resolution in steep-gradient regions without introducing spurious oscillations, while at the same time are of at least second-order accuracy in smooth regions [11-15]. In the past decades, various types of HRS, such as flux-corrected transport (FCT) techniques, total variation diminishing (TVD) schemes, normalized variable diagram (NVD) schemes, and essentially non-oscillatory (ENO) schemes, have been developed [16-20]. In this paper, the widely used high-resolution TVD schemes are considered in the context of the finite volume method (FVM).

TVD schemes, originally developed by Harten [21], are a group of popular high-resolution schemes for solving hyperbolic conservation laws. Depending on some critical conditions, a TVD flux-limiter often switches from a high-order scheme to a low-order diffusive/compressive scheme (or vice versa) for the purpose of circumventing the aforementioned numerical dilemma [22-24]. Particularly, in Ref. [11], a large number of TVD schemes proposed in the literature have been grouped into three broad categories: OTU-TVD (One-step Time-space-coupled Unsteady TVD), MTU-TVD (Multi-step Time-space-separated Unsteady TVD) and SS-TVD (Semi-discrete Steady-state TVD). Their design principles are also presented with a view of choosing different flux-limiter forms for various types of discretization methods (steady or unsteady, time-space-coupled or time-space-separated) in order to achieve better accuracy, convergence and efficiency.

As is well known, TVD schemes, on the basis of a rigorous mathematical foundation, possess several attractive features, such as monotonicity preservation,

computational simplicity and efficiency, and high-order accuracy [25-28]. For structured grids, a number of TVD schemes have been developed in the past decades, such as Albada [11, 13, 16, 24, 29], CUBISTA [11, 13, 30], Harmonic [11-13, 22, 24, 31-34], Koren [11, 13, 23, 24, 34-37], Minmod [11-13, 15, 22, 31, 33, 38-39], MUSCL [11-13, 15-16, 22, 24, 31-32], OSPRE [11, 13, 24], Superbee [11-13, 15, 22, 24, 31, 33, 38], TCDF [11], UMIST [11, 13, 33, 40], and WACEB (also known as the Bounded QUICK limiter) [11, 13, 40-42].

When implementing high-order TVD schemes, the nodal variable values in two upstream cells are required. This poses a difficulty when TVD schemes are extended to arbitrary unstructured grids, because the far-upwind node is not readily available here. It is not straightforward to extend TVD schemes into arbitrary unstructured grids due to this reason, which is especially challenging for three-dimensional problems. Although a large number of TVD schemes have been proposed and analyzed, little attention has been paid to implementing these schemes on arbitrary unstructured grids, partly because of the complexity of locating the far-upstream nodes and determining the variable values there [1-3, 6, 33, 38-39, 43-46].

In this article, several existing r -factor (far-upwind reconstruction) algorithms are reviewed, and their advantages and disadvantages are revealed and analyzed. A refined r -factor algorithm (viz. FFISAM), being suitable for arbitrary unstructured grids, is proposed and benchmarked in several two-dimensional advection test problems. Its performance is compared with several existing algorithms, such as Darwish's and Hou's algorithms [1-3, 6, 33, 38-39, 43-46].

2. TVD Schemes

Most TVD schemes available in the literature can be grouped into three broad categories, viz. OTU-TVD, MTU-TVD and SS-TVD [11]. Different flux-limiter forms (CFL-dependent or CFL-independent, and different limiting conditions) for various types of discretization methods (steady or unsteady, time-space-coupled or time-space-separated) lead to different performance in terms of accuracy and convergence. The extension of these schemes to unstructured meshes can be accomplished by employing a r -factor (far-upwind reconstruction) algorithm.

The focus of this paper is on the extension of TVD schemes to arbitrary unstructured meshes with the aid of the so-called r -factor algorithm, but not the development of TVD schemes themselves. For this purpose, various SS-TVD

schemes, designed for obtaining the steady-state solution of one-dimensional (1D) advection equation [11, 13], are reviewed in this section. The 1D advection equation can be written as:

$$q_t = -(aq)_x \quad (1)$$

where $q = q(x, t)$ denotes the transported variable and a is the advection velocity. Without loss of generality, we assume that the velocity is a positive constant ($a > 0$). The opposite case ($a < 0$) can be handled by symmetry at each cell interface. A pseudo-time stepping technique is employed to get the steady-state solution, and we define Δt as the time step and Δx as the size of the control volume. The flux term in Eq. (1) is deliberately placed on the right-hand side of the equation to emphasize the semi-discrete form. It is obvious that finding the steady-state solution of Eq. (1) is essentially equivalent to solving the steady equation $(aq)_x = 0$.

We would like to emphasize that, although in this paper various r -factor algorithms are discussed and evaluated only based on the SS-TVD schemes, these algorithms are actually equally applicable to other types of TVD schemes (e.g., OTU-TVD, MTU-TVD and other miscellaneous TVD schemes).

2.1. Semi-discrete Steady-state Linear Schemes

Integrating Eq. (1) over a control volume $C_i = [x_{i-\Delta x/2}, x_{i+\Delta x/2}]$, one can obtain the general semi-discrete flux-conservative form:

$$\frac{dq_i}{dt} = \frac{-a(q_{i+1/2} - q_{i-1/2})}{\Delta x} \quad (2)$$

where the interface value, $q_{i+1/2}$, can be obtained by using the well-known k -schemes, firstly introduced by Van Leer [11, 13, 36, 47]:

For uniform grids,

$$q_{i+1/2} = q_i + \frac{1}{2}\psi(\tilde{r}_{i+1/2})(q_i - q_{i-1})$$

$$\text{where } \psi(\tilde{r}_{i+1/2}) = \frac{1+k}{2}\tilde{r}_{i+1/2} + \frac{1-k}{2}, \quad \tilde{r}_{i+1/2} = \frac{(q_{i+1} - q_i)}{(q_i - q_{i-1})} \quad (3)$$

For non-uniform meshes,

$$q_{i+1/2} = q_i + \frac{\Delta x_i}{2}\psi(\tilde{r}_{i+1/2})\left(\frac{\partial q}{\partial x}\right)_{i-1/2}$$

$$\text{where } \psi(\tilde{r}_{i+1/2}) = \frac{1+k}{2}\tilde{r}_{i+1/2} + \frac{1-k}{2}, \quad \tilde{r}_{i+1/2} = \left(\frac{\partial q}{\partial x}\right)_{i+1/2} / \left(\frac{\partial q}{\partial x}\right)_{i-1/2}$$

$$\left(\frac{\partial q}{\partial x}\right)_{i+1/2} = \frac{q_{i+1} - q_i}{x_{i+1} - x_i}, \quad \left(\frac{\partial q}{\partial x}\right)_{i-1/2} = \frac{q_i - q_{i-1}}{x_i - x_{i-1}}, \quad \Delta x_i = x_{i+1/2} - x_{i-1/2} \quad (4)$$

where $\tilde{r}_{i+1/2}$ denotes the ratio of two consecutive gradients: one across the cell face in question $(\partial q / \partial x)_{i+1/2}$ and the other across the immediate upwind interface $(\partial q / \partial x)_{i-1/2}$, which is essentially a local measure of smoothness, and k is a variable representing different schemes. Actually, the k -schemes can be deemed as the sum of a diffusive FOU term and an anti-diffusive term. The anti-diffusive part is multiplied by the flux limiter function $\psi(\tilde{r}_{i+1/2})$, which is a linear function of $\tilde{r}_{i+1/2}$.

As illustrated in Figure 1, some classical linear high-order advection schemes can be derived by taking different values of k , e.g.

$$\begin{aligned} \text{SOU } (k = -1): \quad & \psi(\tilde{r}_{i+1/2}) = 1 \\ \text{Fromm } (k = 0): \quad & \psi(\tilde{r}_{i+1/2}) = (\tilde{r}_{i+1/2} + 1)/2 \\ \text{CUI } (k = 1/3): \quad & \psi(\tilde{r}_{i+1/2}) = (2\tilde{r}_{i+1/2} + 1)/3 \\ \text{QUICK } (k = 1/2): \quad & \psi(\tilde{r}_{i+1/2}) = (3\tilde{r}_{i+1/2} + 1)/4 \\ \text{CDS } (k = 1): \quad & \psi(\tilde{r}_{i+1/2}) = \tilde{r}_{i+1/2} \end{aligned} \quad (5)$$

It is well known that these linear high-order schemes are vulnerable to unphysical spatial oscillations when they are applied to capture shocks or steep variations of the transported variable, primarily due to their unbounded nature [8-10].

It is noted that the semi-discrete flux-limited form can also be formulated in a slightly different form from the one given by Eq. (3) or Eq. (4), as those outlined in [15, 20, 26, 38, 42]. Through simple algebraic manipulations, it is possible to convert those different forms and their corresponding flux-limiters into the usual semi-discrete flux-limited form (Eq. (3) or Eq. (4)), which is the only form considered in this paper for the purpose of consistency.

2.2. Semi-discrete Steady-state TVD Criterion

In order to achieve the desired boundedness for the usual semi-discrete flux-limited form (Eq. (3) or Eq. (4)), several boundedness criteria have been developed in the literature, such as Sweby's TVD criterion [13, 22], Gaskell's CBC criterion [48, 49]

and Spekreijse's positivity criterion [16].

Sweby's TVD criterion, which has been broadly applied to the construction of SS-TVD flux-limiters [11, 13, 15-16, 22, 26, 33-39, 40, 44], is adopted here to construct nonlinear monotonicity-preserving SS-TVD schemes that are capable of suppressing spurious oscillations in the vicinity of discontinuities or steep gradients. As shown in Figure 1, Sweby's TVD criterion is defined as:

$$\begin{cases} 0 \leq \psi(\tilde{r}_{i+1/2}) \leq \min(2\tilde{r}_{i+1/2}, 2) & \text{when } \tilde{r}_{i+1/2} > 0 \\ \psi(\tilde{r}_{i+1/2}) = 0 & \text{when } \tilde{r}_{i+1/2} \leq 0 \end{cases} \quad (6)$$

Obviously, Sweby's TVD criterion (Eq. (6)) is independent from the CFL number. This is an attractive attribute to have for the steady-state solution of the semi-discrete advection equation (Eq. (2)) based on a pseudo-time stepping technique. Note that, although most flux-limiters proposed so far satisfy the above criterion, for negative values of $\tilde{r}_{i+1/2}$, the violation of this criterion does not cause problems in practice, as evidenced in [11, 13-14, 16, 24, 29, 35].

2.3. SS-TVD Flux-limiters

In the past decades, a large number of SS-TVD flux-limiters have been proposed and analyzed. Some of those are outlined below for the convenience of discussion (refer to Figure 2).

1). *Piecewise-linear SS-TVD Flux-limiters:*

Minmod limiter [11-13, 15, 22, 31, 33, 38-39, 46]:

$$\psi(\tilde{r}_{i+1/2}) = \max\left[0, \min(\tilde{r}_{i+1/2}, 1)\right] \quad (7)$$

Superbee limiter [11-13, 15, 22, 24, 31, 33, 38, 46]:

$$\psi(\tilde{r}_{i+1/2}) = \max\left[0, \min(2\tilde{r}_{i+1/2}, 1), \min(\tilde{r}_{i+1/2}, 2)\right] \quad (8)$$

MUSCL limiter [11-13, 15-16, 22, 24, 31-32, 46]:

$$\psi(\tilde{r}_{i+1/2}) = \max\left[0, \min\left(2\tilde{r}_{i+1/2}, \frac{\tilde{r}_{i+1/2} + 1}{2}, 2\right)\right] \quad (9)$$

Koren limiter [11, 13, 23-24, 34-37]:

$$\psi(\tilde{r}_{i+1/2}) = \max\left[0, \min\left(2\tilde{r}_{i+1/2}, \frac{2\tilde{r}_{i+1/2} + 1}{3}, 2\right)\right] \quad (10)$$

WACEB limiter [13, 41] (also known as the Bounded QUICK limiter [40, 42]):

$$\psi(\tilde{r}_{i+1/2}) = \max\left[0, \min\left(2\tilde{r}_{i+1/2}, \frac{3\tilde{r}_{i+1/2} + 1}{4}, 2\right)\right] \quad (11)$$

UMIST limiter [33, 40] (also known as the SPL-1/2 limiter [11, 13]):

$$\psi(\tilde{r}_{i+1/2}) = \max \left[0, \min \left(2\tilde{r}_{i+1/2}, \frac{3\tilde{r}_{i+1/2} + 1}{4}, \frac{\tilde{r}_{i+1/2} + 3}{4}, 2 \right) \right] \quad (12)$$

2). *Gradually-switching Smooth SS-TVD Flux-limiters:*

Harmonic limiter [11-13, 22, 24, 31-34, 46]:

$$\psi(\tilde{r}_{i+1/2}) = \frac{\tilde{r}_{i+1/2} + |\tilde{r}_{i+1/2}|}{\tilde{r}_{i+1/2} + 1} \quad (13)$$

OSPRES limiter [11, 13, 24]:

$$\psi(\tilde{r}_{i+1/2}) = \frac{3\tilde{r}_{i+1/2}(\tilde{r}_{i+1/2} + 1)}{2 \left[(\tilde{r}_{i+1/2})^2 + \tilde{r}_{i+1/2} + 1 \right]} \quad (14)$$

Albada limiter [11, 13, 16, 24, 29] (also known as the GVA-0 limiter [13]):

$$\psi(\tilde{r}_{i+1/2}) = \frac{\tilde{r}_{i+1/2}(\tilde{r}_{i+1/2} + 1)}{(\tilde{r}_{i+1/2})^2 + 1} \quad (15)$$

TCDF limiter [11]:

$$\psi(\tilde{r}_{i+1/2}) = \begin{cases} \tilde{r}_{i+1/2}(\tilde{r}_{i+1/2} + 1) / \left((\tilde{r}_{i+1/2})^2 + 1 \right) & \text{when } \tilde{r}_{i+1/2} < 0 \\ (\tilde{r}_{i+1/2})^3 - 2(\tilde{r}_{i+1/2})^2 + 2\tilde{r}_{i+1/2} & \text{when } 0 \leq \tilde{r}_{i+1/2} < 0.5 \\ 0.75\tilde{r}_{i+1/2} + 0.25 & \text{when } 0.5 \leq \tilde{r}_{i+1/2} < 1.5 \\ \frac{2(\tilde{r}_{i+1/2})^2 - \frac{3}{16} \cdot \tilde{r}_{i+1/2} - \frac{9}{8}}{(\tilde{r}_{i+1/2})^2} & \text{when } 1.5 \leq \tilde{r}_{i+1/2} < +\infty \end{cases} \quad (16)$$

Generally speaking, a piecewise-linear flux-limiter (e.g. Eqs. (7-12)) acts simply as switches between linear schemes and offers the advantage of great flexibility. The overall accuracy of the piecewise-linear flux-limiters can be improved by enlarging the region of a specified high-order scheme (e.g. QUICK or CUI). However, the flux-limiters of this type suffer from an adverse effect on convergence behavior under some circumstances because of their discontinuous nature. In contrast, the smooth flux-limiters (e.g. Eqs. (13-16)) often exhibit better convergence than the piecewise-linear ones at a price of accuracy [11, 13].

In summary, this section provides a brief review of various SS-TVD schemes designed for obtaining the steady-state solution of the 1D advection equation. The key elements of the SS-TVD schemes include the general semi-discrete flux-conservative form (viz. Eq. (2)) to discretize the advection term and the usual semi-

discrete flux-limited form (viz. Eq. (3) or Eq. (4)) to approximate the conservative flux. It is the CFL-independent limiter $\psi(\tilde{r}_{i+1/2})$ that distinguishes different SS-TVD schemes and determines the accuracy of the scheme.

3. Existing r -Factor Algorithms for Unstructured Meshes

As displayed in Figure 3, for a prescribed face of f on an unstructured grid, only the upstream node C and the downstream node D are directly available. Since the index-based notation, which is used above in Eqs. (2-16) for structured grids, is not suitable for unstructured meshes, the more appropriate notation, shown in Figure 3, is adopted here [38]. Apparently, by the definition of an imaginary far-upwind node U , which is an upwind node of C , the usual semi-discrete flux-limited form can be rewritten as the following expressions:

For uniform meshes,

$$q_f = q_C + \frac{1}{2} \psi(\tilde{r}_f)(q_C - q_U), \quad \text{where } \tilde{r}_f = \frac{(q_D - q_C)}{(q_C - q_U)} \quad (17)$$

For non-uniform meshes,

$$q_f = q_C + \frac{\Delta x_C}{2} \psi(\tilde{r}_f) \frac{q_C - q_U}{x_C - x_U}$$

$$\text{where } \tilde{r}_f = \left(\frac{q_D - q_C}{x_D - x_C} \right) / \left(\frac{q_C - q_U}{x_C - x_U} \right), \quad \Delta x_C = 2 \cdot (x_f - x_C) \quad (18)$$

where $\psi(\tilde{r}_f)$ is the CFL-independent limiter (e.g., Eqs. (7-16)). As can be seen from Eqs. (17-18), the far-upwind node location x_U and variable value q_U are required for the calculation of the face value q_f . However, on arbitrary unstructured grids, the far-upwind node U is not readily available and thus it is not straightforward to implement these SS-TVD schemes on unstructured grids.

We would like to reemphasize that, although various far-upwind reconstruction techniques (viz. r -factor algorithms) are presented and discussed here only in the context of the SS-TVD schemes, these algorithms can be equally applied to other types of TVD schemes (e.g., OTU-TVD, MTU-TVD and other miscellaneous TVD schemes), which are not taken into account in this paper.

3.1. Extrapolation r -Factor Algorithms

3.1.1. Bruner's r -Factor Algorithm

Since $du=2d_f$ on a uniform one-dimensional grid (du is the magnitude of the vector between the nodes U and C , and d_f is that between the node C and the center of the face f), Bruner [50] proposed the following modification to the definition of the r -factor for implementing various TVD schemes on an arbitrary unstructured grid:

$$\tilde{r}_f = \frac{q_D - q_C}{q_C - q_U} \approx \frac{q_D - q_C}{2(q_f - q_C)} = \frac{q_D - q_C}{(\nabla q)_C \cdot \vec{d} \cdot 2d_f} \quad (19)$$

where \vec{d} is the unit vector from C to D , and $(\nabla q)_C$ represents the gradient of the variable q at the center of the donor cell C .

As pointed out by Darwish and Moukalled [38], Bruner's r -factor algorithm is inconsistent because, when applied to one-dimensional meshes, it does not recover to the original TVD formulation:

$$\tilde{r}_f = \frac{q_D - q_C}{q_C - q_U} \neq \frac{q_D - q_C}{(\nabla q)_C \cdot \vec{d} \cdot 2d_f} \quad (20)$$

3.1.2. Darwish's r -Factor Algorithm

It is apparent that the main difficulty in implementing TVD schemes on an arbitrary unstructured mesh lies in the need for defining a virtual far-upwind node U . Darwish and Moukalled [38] proposed the so-called exact r -factor technique, which is broadly used to resolve this issue in the literature [1, 2, 6, 33, 38-39, 43-46]. Darwish's r -factor formulation can be written as:

$$\tilde{r}_f = \frac{q_D - q_C}{q_C - q_U} = \frac{q_D - q_C}{q_D + (q_C - q_U) - q_D} = \frac{q_D - q_C}{(q_D - q_U) - (q_D - q_C)} \quad (21)$$

where q_D and q_C denote the variable values at the nodes that straddle the considered interface and thus are readily available for arbitrary unstructured grids. In order to compute the term involving q_U , it was assumed that:

$$\begin{aligned} (\nabla q)_C \cdot \vec{d}(du + dd) &= (\nabla q)_C \cdot \vec{d} \cdot 2dd = q_D - q_U \Rightarrow \\ q_U &= q_D - (\nabla q)_C \cdot \vec{d} \cdot 2dd \end{aligned} \quad (22)$$

where \vec{d} is the unit vector from C to D , du is the magnitude of the vector between the nodes U and C , dd is the magnitude of the vector between the nodes C and D , and q_U is the predicted value at the virtual far-upwind node, as shown in Figure 3.

Essentially, Darwish's r -factor algorithm extends various TVD schemes to arbitrary unstructured grids by constructing a virtual far-upwind node U and further using the 'equal-distance' and 'equal-gradient' assumptions [1, 2, 6, 38]. Under these

assumptions, the distance between U and C , du , is supposed to be equal to that between C and D , dd . Besides, it assumes that the variable q uniformly changes from U to D , which means that the gradient ∇q keeps as a constant along the whole segment (UD). It can be proved that Eq. (22) can be directly derived from the aforementioned two assumptions.

3.1.3. Zhang's Far-upwind Reconstruction Technique

Zhang et al. [2, 6] debated that Darwish's far-upwind reconstruction technique has two inherent drawbacks. Firstly, the imaginary node U may be located several cells away from the donor cell C , when extremely non-uniform grids are adopted. Secondly, it is not all that appropriate to assume that the gradient ∇q stays the same along the whole segment UD . In order to overcome the aforementioned two drawbacks, New-Technique-2 is proposed and evaluated in their paper, where a predictor-corrector procedure is employed to determine the value of du and a parabolic function is used to represent the variable variation between the nodes U and D on the basis of q_C , q_D and $(\nabla q)_C$.

Ref. [6] clearly shows that this technique results in a better overall performance in terms of accuracy and convergence for many NVD schemes, when compared with Darwish's technique (Eq. (22)). Since the New-Technique-2 is originally designed for NVD schemes, the implementation of this algorithm in TVD schemes has not been reported and is beyond the scope of this study.

3.2. Interpolation r -Factor Algorithms

3.2.1. Li's r -Factor Algorithm

Li et al. [33, 44, 45] pointed out that the value of q_U in Darwish's r -factor algorithm is essentially extrapolated by the variable value of the downstream node D (q_D) and the gradient value at the upstream node C ($(\nabla q)_C$), as demonstrated in Eq. (22). With the aid of Gauss's theorem, $(\nabla q)_C$ can be achieved by interpolating the q values on all faces of the donor cell C , which is the only process that concerns the far-upwind information. Furthermore, Li has demonstrated that, if a parabolic distribution of q along nodes U , C and D in one-dimensional grid is assumed, q_U can be exactly estimated by Eq. (22). However, if the distribution of q is exponential as in some applications, Eq. (22) will lead to an undesirable or even an erroneous estimation of

q_U . In view of this, Li [44] proposed a scheme that estimates the value of q_U by interpolation rather than extrapolation.

In Li's reconstruction method [44], the same 'equal-distance' assumption as that of Darwish's algorithm is employed, which is essentially equivalent to that the virtual far-upwind node U is determined in such a way that it lies along the line joining nodes D and C , with C at the center of the UD segment. In addition, the gradient of the node U_r , instead of the node C , is used to estimate the value of q_U in Eq. (21).

$$q_U = q_{U_r} + (\nabla q)_{U_r} \cdot \vec{d}_{U_r, U} \quad (23)$$

where U_r is the centroid of the cell which contains the virtual far-upwind node U , $(\nabla q)_{U_r}$ indicates the gradient of U_r , and $\vec{d}_{U_r, U}$ stands for the vector between the nodes U_r and U , as illustrated in Figure 4.

Li et al. [44] stated that this algorithm allows more upwind information to be included. Especially, when the distributions of q_U , q_C and q_D are not parabolic (e.g. obeying an exponential distribution), Darwish's extrapolation algorithm (Eq. (22)) leads to an undesirable prediction of q_U , but Li's interpolation algorithm (Eq. (23)) works reasonably well. Additionally, it should be pointed out that the search process for U_r is actually very complex, because different mathematical algorithms are needed for various kinds of polygonal grids (e.g. triangle, tetrahedron, quadrangle, hexahedron, pentagon, and so on.), which consequently prevent this technique from being widely used on arbitrary unstructured grids.

3.2.2. Hou's r -Factor Algorithm

Given that the flux through the interface depends primarily on the normal variable transportation, Hou et al. [33, 45] and Zhang et al. [2, 6] pointed out that, as demonstrated in Figure 5, for the interface f between C and D , the face value q_f should be calculated using auxiliary nodes C' and D' , which are the projections of C and D , respectively, along the line L that runs through the center of the interface and perpendicular to it. Moreover, in the discretization of the convection term, the differences between the physical quantities at D' and D (or C' and C) are ignored. This is essentially equivalent to assuming that the node D' (or C') has the same value and gradient as the node D (or C), i.e., $q_D = q_{D'}$, $q_C = q_{C'}$, $(\nabla q)_D = (\nabla q)_{D'}$ and $(\nabla q)_C = (\nabla q)_{C'}$.

As shown in Figure 5, in Hou's far-upwind reconstruction technique [33], the

imaginary far-upwind node U is defined as the intersection of the line L and the line l , which passes through U_r (the closest centroid of the further upwind cells) and is perpendicular to the normal of the interface. Here, U_r is selected by comparing the distances between the line L and the centroids of all cells adjacent to the opposite vertex P_2 except for the donor cell C .

Evidently, the variable values at the nodes U , C' and D' can be obtained by interpolation from the values at the cell-centers U_r , C and D , respectively. But, for simplicity and efficiency, Hou et al. [33] suggests to directly replace the former by the latter. Consequently, Eq. (18) can be reformulated as:

$$q_f = q_C + \frac{\Delta x_{C'}}{2} \psi(\tilde{r}_f) \frac{q_C - q_{U_r}}{x_{C'} - x_U}$$

$$\text{where } \tilde{r}_f = \left(\frac{q_D - q_C}{x_{D'} - x_{C'}} \right) \Bigg/ \left(\frac{q_C - q_{U_r}}{x_{C'} - x_U} \right), \quad \Delta x_{C'} = 2(x_f - x_{C'}) \quad (24)$$

Numerical results presented in [33, 45] confirm that Hou's r -factor algorithm (Eq. (24)) performs better than Li's (Eq. (23)) and Darwish's (Eq. (22)) algorithms because it results in higher accuracy and less oscillation in steep gradient regions, especially for low quality two-dimensional (2D) unstructured meshes. Since Hou's algorithm was derived only based on 2D unstructured triangular grids, it needs to be further improved to be applicable to arbitrary unstructured grids.

4. A Refined r -Factor Algorithm for Arbitrary Unstructured Meshes

This section presents a further analysis of the aforementioned existing r -factor algorithms, and theoretically demonstrates the advantages and disadvantages of these algorithms. Furthermore, based on the analysis, a refined r -factor algorithm, referred to as FFISAM, is proposed and evaluated in order to more efficiently extend TVD schemes to arbitrary unstructured meshes.

4.1. Further Analysis of Existing r -Factor Algorithms

4.1.1. Darwish's r -Factor Algorithm

Despite the well-known Darwish's r -factor algorithm is widely used in the literature [1, 2, 6, 33, 38-39, 43-45], a further analysis of this far-upwind reconstruction technique reveals the following issues that need to be resolved:

1). As aforementioned, it is apparent that the flux through the interface depends primarily on the normal variable transportation. As a consequence, the variables in the effective advection direction (normal to the interface) should have been taken into account in calculating the flux. In fact, as shown in Figure 6, in Darwish's algorithm (Eq. 22), the flux is solely determined by the variables along the line joining the centroids of the upwind and downwind cells (C and D). This is not physically appropriate for non-orthogonal grids and may be a source of numerical errors.

2). To obtain the far-upwind node variable value q_U , this technique assumes that the gradient ∇q does not change along the whole segment (UD). Consequently, the value at the center of the upwind cell (q_C) can be written as:

$$q_C = q_D - (\nabla q)_C \cdot \vec{CD} \quad (25)$$

Obviously, this is contradictory to the real physical situation, because q_C is an unknown fixed value before the new pseudo-time step.

3). Although Eq. (22) is able to accurately simulate parabolic distributions of q along the cells U , C and D , when the exponential distribution is considered, Eq. (22) will result in undesirable or even erroneous estimations for the value of q_U [33, 44-45]. Clearly, improved reconstruction techniques are required, either by extrapolation or interpolation.

4.1.2. Li's r -Factor Algorithm

In Li's r -factor algorithm [44], the gradient of node U_r (the center of the cell containing the virtual node U) instead of node C is used to estimate the far-upwind node variable value q_U . This suggests that this technique obtains the value of q_U by interpolation rather than extrapolation, which is an improvement over Darwish's algorithm. However, the following aspects could possibly be further improved:

1). Although Li's algorithm overcomes the aforementioned second and third drawbacks of Darwish's algorithm to a certain degree, the first drawback of the latter still exist in the former.

2). As discussed in Section 4.2, since the cell containing the virtual node U is not guaranteed to be the closest centroid of the far-upwind cells. The way of defining the node U_r could potentially be further improved for this algorithm. For instance, as shown in Figure 6, the cell H is closer to the virtual node U than the cell I containing the node U . The use of H could lead to a better approximation of the value of q_U .

3). Li's algorithm was originally developed for inner cells, and therefore a special treatment or formulation may be required for boundary cells. Otherwise, the virtual node U may be located outside of the computational domain, and thus the cell containing the node U cannot be identified, viz. the node U_r does not exist in this situation, as displayed in Figure 7.

4). In addition, although it is convenient to implement Li's algorithm on triangular or quadrilateral grids, its application to arbitrary unstructured grids, consisting of hexahedrons, pentagons and hexagons, may be complicated and time-consuming, as different mathematical algorithms are required for various types of polygonal unstructured grids.

4.1.3. Hou's r -Factor Algorithm

Hou's r -factor algorithm [33, 45] overcomes all the aforementioned three drawbacks of Darwish's algorithm, but it is developed specifically for 2D triangular grids. The extension of this algorithm to arbitrary unstructured grids has not been attempted yet.

4.2. A Refined r -Factor Algorithm

The r -factor algorithm proposed in this paper (viz. FFISAM) is essentially an extension of Li's and Hou's interpolation algorithms that retains the attractive attributes of the both algorithms and overcomes the issues identified above. In FFISAM, the following equation is employed:

$$q_f = q_{C'} + \frac{\Delta x_{C'}}{2} \psi(\tilde{r}_f) \frac{q_{C'} - q_U}{x_{C'} - x_U}, \quad q_U = q_{U_r} + \vec{d}_{U,U} \cdot (\nabla q)_{U_r}$$

$$\text{where } \tilde{r}_f = \left(\frac{q_{D'} - q_{C'}}{x_{D'} - x_{C'}} \right) \bigg/ \left(\frac{q_{C'} - q_U}{x_{C'} - x_U} \right), \quad q_{D'} = q_D, \quad q_{C'} = q_C, \quad \Delta x_{C'} = 2 \cdot (x_f - x_{C'}) \quad (26)$$

where $\psi(\tilde{r}_f)$ is the CFL-independent flux limiter (e.g., Eqs. (7-16)). U_r is defined as the closest effective far-upwind cell, which is determined in a different way from that in Li's (Eq. (23)) and Hou's (Eq. (24)) algorithms. $(\nabla q)_{U_r}$ represents the gradient of U_r and can be obtained with the aid of Gauss's theorem. U is the virtual far-upwind node, and $\vec{d}_{U,U}$ denotes the vector between the centroid of U_r and the node U . All the above definitions will be described in detail in the following paragraphs with the help of Figure 6 and Figure 7.

- For the Inner Domain

As demonstrated in Figure 6, for the interface ab between C and D on an arbitrary unstructured mesh, considering the fact that the flux through the interface depends principally on the normal variable transportation, the face value q_f should be calculated using auxiliary nodes C' and D' , which are the projections of C and D , respectively, along the line L that cuts through the center of the considered interface (ab) perpendicularly. As aforementioned in Section 3.2.2, for the discretization of the convection term, the differences between the physical quantities at D' and D (or C' and C) can be neglected, which means that the node D' (or C') has the same variable value as the node D (or C), viz. $q_D = q_{D'}$ and $q_C = q_{C'}$, as shown in Eq. (26).

Furthermore, similar to Darwish's and Li's algorithms, the 'equal-distance' assumption is employed in FFISAM in order to determine the location of the virtual far-upwind node U . However, as illustrated in Figure 6, here the virtual node U is determined in such a way that it lies along the line joining nodes D' and C' , with C' at the center of the UD' segment (viz. $du=dd$), which is slightly different from the way in Darwish's and Li's algorithms.

Additionally, to obtain the value of q_U , all the upwind-cell's nodes are taken into account except for the two end nodes of the chosen face (a, b), viz. the nodes c, d, e and f in Figure 6. Then, the closest effective far-upwind cell U_r is identified by comparing the distances between the virtual node U and the centroids of all cells adjacent to the aforementioned considered nodes (except for the upwind cell C itself), such as B, E, F, G, H, I and J in Figure 6. After identifying the cell U_r , Eq. (26) can be used to calculate the value of q_U and eventually the face value q_f .

It should be emphasized that the cell U_r of FFISAM is not always the same as the cell U_r of Li's algorithm, which contains the virtual far-upwind node U . As evidenced in Figure 6, the cell H is selected as the U_r for the former, however, the cell I is chosen as the U_r for the latter. It is speculated that FFISAM is more preferable in this situation, because the cell H is closer to the virtual node U than the cell I , and the use of H can lead to a better approximation of the value of q_U .

- For the Boundary Domain

When it comes to the boundary domain, some minor corrections to the above inner-domain's far-upwind reconstruction procedure are necessary to obtain a reasonable estimation of the value of q_U .

As illustrated in Figure 7, for the interface ab between C and D , the same ‘equal-distance’ assumption as the inner-domain case is adopted to determine the location of the virtual node U . If the above inner-domain’s reconstruction procedure was followed, cell A would have been selected as the closest effective far-upwind cell U_r , because only the cell A is adjacent to the node c , which is the upwind-cell’s unique node except for the two end nodes of the considered face (a, b). However, under this circumstance, it is not so advisable to estimate the value of q_U by using the variable value and gradient of A , because the centroid of A is far away from the virtual node U . The adoption of the centroid of the upwind-cell’s boundary face bc , viz. the node d , may lead to a better approximation of q_U . On this basis, it is suggested that U_r should be chosen from the centroid of the closest effective far-upwind cell (following the procedure in the inner-domain case) and the centroids of the upwind-cell’s boundary faces by comparing the distances between these centroids and the virtual node U . Moreover, if the centroid of the upwind-cell’s boundary face is selected as U_r , to render FFISAM more efficient, the value of q_U is simply estimated as the variable value of this centroid (e.g., $q_U = q_d$ in Figure 7.)

It deserves our attention that, if Hou’s algorithm was used in Figure 7 (following the procedure in Section 3.2.2), the cell A would have been chosen as the cell U_r , and the node A' would have been defined as the virtual far-upwind node U . Apparently, this would have been contradictory to the definition of the node U , in view of the fact that the node A' is actually in the downstream of the centroid of the upwind-cell C . Besides, none of the cells contain the node U in this case, because it lies outside of the computational domain as displayed in Figure 7. Hence, it can be concluded that Li’s and Hou’s algorithms also require different formulations for boundary cells in order to be applicable in all situations.

To ensure the physical boundedness of q_U , all the neighboring cells and boundary faces of the up-wind cell C are considered except for the down-wind cell D , such as B, E, F, G, H, I and J in Figure 6 and the boundary face bc in Figure 7. The following equation is employed to restrict the value of q_U .

$$q_U = \min \left\{ \max \{ q_U, q_C^{\min} \}, q_C^{\max} \right\} \quad (27)$$

where q_C^{\min} and q_C^{\max} respectively represent the maximum and minimum values of the aforementioned considered neighboring cells, boundary faces and the cell C itself.

With the aid of FFISAM (Eq. (26) and Eq. (27)), the implementation of TVD schemes on arbitrary unstructured meshes is now simple and straightforward even for

three-dimensional problems, because the value of q_f can be calculated in the same way for all the faces irrespective of the dimensionality of control volumes and the topological structure of unstructured meshes.

The attractive attributes of FFISAM are summarized. First of all, the face position can be taken into consideration for non-uniform unstructured grids as shown by Eq. (26). Secondly, similar to Li's (Eq. (23)) and Hou's (Eq. (24)) algorithms, FFISAM obtains the value of the imaginary far-upwind node (q_U) by interpolation rather than extrapolation. This allows more upwind information to be included relative to Bruner's (Eq. (19)) and Darwish's (Eq. (22)) extrapolation algorithms. Thirdly, the use of FFISAM facilitates the extension of TVD schemes to arbitrary unstructured meshes without having to deal with some issues as in Li's and Hou's algorithms discussed earlier on. The last but not the least, the implementation of FFISAM is simple and straightforward, and does not involve a complex and time-consuming search procedure for U_r . Based on these, it is reasonable for us to believe that FFISAM offers improvements over the existing r -factor algorithms in terms of numerical properties. A number of numerical tests will be presented in the next section to validate FFISAM.

5. Numerical Test Cases

This section presents several classic numerical examples to evaluate the relative performance of the above r -factor algorithms for ten classical SS-TVD flux-limiters, including Albada, Harmonic, Koren, Minmod, MUSCL, OSPRE, Superbee, UMIST, WACEB and TCDF. For arbitrary unstructured meshes, only Darwish's algorithm (Eq. (21) and Eq. (22)) and the newly-proposed FFISAM (Eq. (26) and Eq. (27)) are compared, because Hou's and Li's algorithms are not suitable for arbitrary unstructured meshes. As aforementioned, since Hou's algorithm was originally designed for triangular or tetrahedral grids, it is included in comparison in Test 4 where both an arbitrary unstructured mesh and an unstructured triangular grid are employed. As for Li's algorithm, since the work concerning the search of U_r (the centroid of the cell containing the virtual node U) is prohibitively complex and time-consuming for arbitrary polygonal unstructured grids (e.g. triangle, tetrahedron, quadrangle, hexahedron, pentagon, and so on.), it is not included in the comparison.

Prescribed velocity fields are used and the accuracy is quantitatively measured by the average L_2 norm of the difference between the exact and numerical solutions:

$$E = \frac{1}{N} \cdot \text{Sqrt}(\sum_{i=1}^N (q_i^n - q_i^a)^2) \quad (28)$$

where q_i^n is the calculated solution after n time steps, q_i^a the exact analytical solution and N the number of control volumes [1, 2, 6, 11].

5.1. Test 1: Pure convection of a step profile

In this problem we consider the advection of a passive scalar in an oblique uniform velocity field of $(u, v)=(1, 1)$. The governing conservation equation of this problem is:

$$\frac{\partial q}{\partial t} + \frac{\partial(uq)}{\partial x} + \frac{\partial(vq)}{\partial y} = 0 \quad (29)$$

where q is the transported variable, and $u=1$ and $v=1$ are the Cartesian components of the given velocity vector. As shown in Figure 8, the inlet boundary conditions are prescribed as:

$$\begin{aligned} q(0, y) &= 1 & \text{for } 0 \leq y \leq 1 \\ q(x, 0) &= 0 & \text{for } 0 < x \leq 1 \end{aligned} \quad (30)$$

The step profile provides the most stringent gradient variation, with the purpose of evaluating the method's ability to resolve a sharp front, with minimum numerical diffusion and without oscillations [6, 11, 30].

An arbitrary unstructured mesh composed of 4057 cells is employed in this test, as displayed in Figure 9. To reach the steady state, a pseudo-time stepping approach is adopted and computations are performed with four different time steps, yielding maximum Courant numbers (Cu) of 0.5, 1.0, 1.5 and 2.0, respectively. Moreover, the first-order implicit Euler method is used for the time discretization because it allows large time steps to be taken, which is especially useful for solving steady flow problems. Specifically, the FOU value (q_c) in Eq. (18) or Eq. (26) is implicitly discretized and the high-order correction term is explicitly treated as the source term.

Table 1 summarizes the errors of Darwish's algorithm and FFISAM for the aforementioned ten classical SS-TVD flux-limiters along the line $x=0.8$ when the steady-state solution is achieved. Figures 10-19 compare the resulting profiles of q along $x=0.8$ and the decay of the residuals at a Courant number of 1.0 for each considered TVD scheme and for different r -factor algorithms. It should be pointed out that, for the above four different Courant numbers, almost the same steady-state solution and convergence conclusions can be reached, therefore, only the numerical results at a

Courant number of 1.0 are demonstrated for conciseness of presentation.

Obviously, for Koren, Superbee, MUSCL, WACEB and UMIST, when compared with Darwish's algorithm, FFISAM does result in significant improvement in accuracy and convergence, as evidenced in Table 1 and Figures 10-14. Special attention needs to be paid to the well-known Superbee scheme, which is known as one of the most compressive differencing schemes and does produce the most accurate result in this test, as illustrated in Figure 11. Moreover, the combination of Superbee with FFISAM produces a much improved profile over the widely-used Darwish's algorithm.

In addition, as shown in Figures 15-19, comparisons of the numerical solutions, obtained with various r -factor algorithms, indicate that FFISAM gives more accurate predictions and slightly better convergence properties when compared to Darwish's algorithm for OSPRE, Albada, Harmonic, Minmod and TCDF.

In summary, the numerical results clearly show that the newly-proposed FFISAM leads to a better performance relative to the widely-used Darwish's algorithm in terms of accuracy and convergence for all the considered ten SS-TVD flux-limiters in this test, which agrees with the theoretical analysis in the previous sections.

5.2. Test 2: Pure convection of a sine-square profile

As illustrated in Figure 20, the same physical domain, governing equation, time discretization scheme and velocity field as the step-profile case are adopted in conducting the advection test of a sine-square profile. The inlet boundary conditions are given as:

$$\begin{aligned}
 q(0, y) &= \sin^2\left(\frac{10}{3}\pi y\right) & \text{for } 0 \leq y \leq \frac{3}{10} \\
 q(0, y) &= 0 & \text{for } 3/10 < y \leq 1 \\
 q(x, 0) &= 0 & \text{for } 0 < x \leq 1
 \end{aligned} \tag{31}$$

The sine-square is a relatively steep profile that is often used to benchmark various flux-limiters in steep gradient regions. In this test, an arbitrary unstructured mesh composed of 5728 cells is employed, as depicted in Figure 21. Computations are performed with four different time steps as well, yielding maximum Courant numbers of 0.5, 1.0, 1.5 and 2.0, respectively.

The errors of various r -factor algorithms for the ten classical SS-TVD flux-limiters when the steady-state is reached are also demonstrated in Table 1. As

expected, almost the same conclusions as the step-profile case can be deduced from the numerical results. However, for compactness of presentation, only the resulting profiles of q along $x=0.6$ and the corresponding residual errors ($Cu=1.0$) for seven TVD schemes (Koren, Superbee, MUSCL, WACEB, TCDF, OSPRE and Albada) are shown in Figures 22-28. Again, FFISAM exhibits better overall performance than Darwish's algorithm in terms of accuracy and convergence.

5.3. Test 3: Pure convection of a semi-ellipse profile

In the third test, we consider the advection of a semi-ellipse profile, and, as illustrated in Figure 29, the inlet boundary conditions are set as:

$$\begin{aligned} q(0, y) &= \sqrt{1 - [y / (1/6)]^2} & \text{for } |y| \leq 1/6 \\ q(0, y) &= 0 & \text{elsewhere} \end{aligned}$$

and

$$\begin{aligned} q(x, 0) &= \sqrt{1 - [x / (1/6)]^2} & \text{for } |x| \leq 1/6 \\ q(x, 0) &= 0 & \text{elsewhere} \end{aligned} \quad (32)$$

The same physical domain, governing equation, velocity vector and time discretization scheme as the first test case are employed. Besides, an arbitrary unstructured mesh of 5249 cells is used, as shown in Figure 30. Four time steps are adopted which correspond to the maximum Courant numbers of 0.5, 1.0, 1.5 and 2.0, respectively.

After reaching the steady-state, the errors along $x=0.8$ for different flux-limiters are summarized in Table 1. The resulting profiles of q along this line and the corresponding residual errors ($Cu=1.0$) are given in Figures 31-37 for seven of the ten TVD schemes that are compared in this study. The performance of FFISAM in other three schemes is similar to the seven schemes shown here.

It is interesting to observe that the Superbee scheme does tend to compress the relatively smooth semi-ellipse profile into a step-like profile due to its inherent squaring effect, as evidenced in Figure 32. It appears that FFISAM introduces more severe squaring-effect than Darwish's algorithm for this TVD flux-limiter. This is likely the reason for the smaller error of Darwish's algorithm than FFISAM shown in Table 1 for this case, in contrast to previous two test cases where FFISAM provides better predictions than Darwish's algorithm when the Superbee scheme is used.

Except for the above difference, almost the same conclusions as Test 1 and Test 2

can be obtained in this test. As with previous tests, FFISAM demonstrates excellent overall performance for the ten SS-TVD flux-limiters relative to Darwish's algorithm in terms of accuracy and convergence.

5.4. Test 4: Pure convection in a rotational velocity field

The former three tests show an overall better performance of FFISAM compared to Darwish's algorithm when the velocity field is uniform. To evaluate the ability of FFISAM in non-uniform flow fields, the advection of a double-step profile in a rotational velocity field is simulated in Test 4. As shown in Figure 38, the inlet boundary conditions are prescribed as:

$$\begin{aligned} q(x,0) &= 1 && \text{for } -0.8 \leq x \leq -0.6 \\ q(x,0) &= 0 && \text{for } -0.6 \leq x \leq 0 \text{ and } -1 \leq x \leq -0.8 \end{aligned} \quad (33)$$

The double-step profile is transported clockwise from the inlet boundary ($x < 0, y = 0$) to the outlet boundary ($x > 0, y = 0$) by a rotational velocity field defined as:

$$U = y, \quad V = -x \quad (34)$$

As shown in Figure 39, this problem is firstly solved on an arbitrary unstructured grid composed of 6428 cells at four different time steps, yielding maximum Courant numbers of 0.5, 1.0, 1.5 and 2.0, respectively. The same governing equation and time discretization scheme as Test 1 are selected in this test. The accuracy is quantitatively measured by the average L_2 norm at the outlet plane ($0 \leq x \leq 1.0, y = 0$), and Table 1 summarizes the accuracy of different r -factor algorithms for the ten TVD schemes.

In fact, almost the same conclusions as in the previous tests can be drawn here for different r -factor algorithms (FFISAM and Darwish's). The resulting profiles of q along the outlet plane and the corresponding residual errors ($\text{Cu} = 1.0$) are demonstrated in Figures 40-45 for six of the ten TVD schemes that are compared in this study. The performance of FFISAM in other four schemes is similar to the six schemes shown here.

The performance of FFISAM is deliberately tested on an unstructured triangular mesh in order to compare with Hou's algorithm, because Hou's algorithm is only applicable to unstructured triangular or tetrahedral grids. As illustrated in Figure 46, in Test 4, calculations are performed on an unstructured triangular mesh consisting of 8682 cells. Table 2 summarizes the errors of different r -factor algorithms (Darwish's, Hou's and FFISAM) for all the ten SS-TVD schemes.

As expected, FFISAM provides the most accurate predictions among all the three r -factor algorithms, and Hou's algorithm gives slightly better performances than Darwish's algorithm, which is in line with the conclusions in Ref. [33]. Besides, Figures 47-52 exhibit the resulting profiles of q along the outlet plane and the corresponding convergence processes ($Cu=1.5$) for six TVD schemes (Koren, Superbee, MUSCL, WACEB, TCDF and Harmonic). It is obvious that FFISAM shows the rapidest decay of the residuals among all the three algorithms in these figures. Actually, when it comes to the other four SS-TVD schemes, the same conclusions can be drawn as well, although these are not presented for the compactness of the paper.

In conclusion, even in this more complicated situation, whether on arbitrary unstructured grids or on triangular grids, the numerical results also demonstrate the apparent advantages of FFISAM over the other r -factor algorithms for the ten SS-TVD schemes in terms of accuracy and convergence.

6. CONCLUSIONS

In this article, the details of several existing r -factor (viz. far-upwind reconstruction) algorithms are reviewed from the perspective of extending TVD schemes to arbitrary unstructured grids. Their respective advantages and disadvantages are also revealed and analyzed. Based on the review, a refined r -factor algorithm, referred to as the FFISAM, is developed, which overcomes some of the inherent drawbacks of the existing algorithms. In the FFISAM, the variables in the effective advection direction (viz. normal to the interface) are chosen for calculating the flux, which is believed to be physically sound. Moreover, since the FFISAM determines the virtual far-upwind node value q_U by interpolation rather than extrapolation, it allows more upwind information to be included and consequently gives more accurate predictions than Darwish's algorithm in some cases, e.g., with an exponential distribution of the advected quantity. In addition, the FFISAM ensures a reasonable virtual far-upwind node location on arbitrary unstructured grids. Four advection tests are conducted to evaluate the relative performance of various r -factor algorithms for ten classical SS-TVD schemes, including Albada, Harmonic, Koren, Minmod, MUSCL, OSPRE, Superbee, UMIST, WACEB and TCDF. The numerical results demonstrate that the FFISAM leads to a higher accuracy and faster convergence than the existing algorithms on arbitrary unstructured meshes.

ACKNOWLEDGEMENTS

The authors gratefully acknowledge the financial support provided by the National Natural Science Foundation of China (Grant No. 51279082) and the support from Australian Research Council through a Discovery Grant (Project ID: DP110105171).

REFERENCES

1. O. Ubbink, R.I. Issa. A method for capturing sharp fluid interfaces on arbitrary meshes. *Journal of Computational Physics*; **153** (1) (1999), 26-50.
2. D. Zhang, C. Jiang, D. Liang, Z. Chen, Y. Yang, Y. Shi, A refined volume-of-fluid algorithm for capturing sharp fluid interfaces on arbitrary meshes. *Journal of Computational Physics*, **274** (2014), 709-736.
3. F. Denner, B.G.M. van Wachem. Compressive VOF method with skewness correction to capture sharp interfaces on arbitrary meshes. *Journal of Computational Physics*, **279** (2014), 127-144.
4. R. Scardovelli, S. Zaleski. Direct numerical simulation of free-surface and interfacial flow. *Annual Review of Fluid Mechanics*, **31** (1) (1999), 567-603.
5. B.P. Leonard, J.E. Drummond. Why you should not use ‘Hybrid’, ‘Power-Law’ or related exponential schemes for convective modelling-there are much better alternatives. *International Journal for Numerical Methods in Fluids*, **20** (6) (1995), 421-442.
6. D. Zhang, C. Jiang, C. Yang, Y. Yang. Assessment of different reconstruction techniques for implementing the NVSF schemes on unstructured meshes. *International Journal for Numerical Methods in Fluids*, **74** (3) (2014), 189-221.
7. W. Shyy. A study of finite difference approximations to steady-state, convection-dominated flow problems. *Journal of Computational Physics*, **57** (3) (1985), 415-438.
8. T. Hayase, J.A.C. Humphrey, R. Greif. A consistently formulated QUICK scheme for fast and stable convergence using finite-volume iterative calculation procedures. *Journal of Computational Physics*, **98** (1) (1992), 108-118.
9. D.B. Spalding. A novel finite difference formulation for differential expressions involving both first and second derivatives. *International Journal for Numerical Methods in Engineering*, **4** (4) (1972), 551-559.
10. P. Tamamidis, D.N. Assanis. Evaluation of various high-order-accuracy schemes with and without flux limiters. *International Journal for Numerical Methods in Fluids*, **16** (10) (1993), 931-948.
11. D. Zhang, C. Jiang, D. Liang, L. Cheng. A review on TVD schemes and a refined flux-limiter for steady-state calculations. *Submitted to Journal of Computational Physics*, (2015).
12. R.J. LeVeque. High-resolution conservative algorithms for advection in incompressible flow. *SIAM Journal on Numerical Analysis*, **33** (2) (1996), 627-665.
13. N.P. Waterson, H. Deconinck. Design principles for bounded higher-order convection

- schemes—a unified approach. *Journal of Computational Physics*, **224**(1) (2007), 182-207.
14. V. Daru, C. Tenaud. High order one-step monotonicity-preserving schemes for unsteady compressible flow calculations. *Journal of Computational Physics*, **193** (2) (2004), 563-594.
 15. Y. Wang, K. Hutter. Comparisons of numerical methods with respect to convectively dominated problems. *International journal for numerical methods in fluids*, **37** (6) (2001), 721-745.
 16. V. Venkatakrishnan. Convergence to steady state solutions of the Euler equations on unstructured grids with limiters. *Journal of Computational Physics*, **118** (1) (1995), 120-130.
 17. G.S. Jiang, C.W. Shu. Efficient implementation of weighted ENO schemes. *Journal of Computational Physics*, **126** (1) (1996), 202-228.
 18. J.P. Boris, D.L. Book. Flux-corrected transport. I. SHASTA, A fluid transport algorithm that works. *Journal of Computational Physics*, **11** (1) (1973), 38-69.
 19. S.T. Zalesak, Fully multidimensional flux-corrected transport algorithms for fluids. *Journal of Computational Physics*, **31** (3) (1979), 335-362.
 20. M.S. Darwish, F.H. Moukalled. Normalized variable and space formulation methodology for high-resolution schemes. *Numerical Heat Transfer*, **26** (1) (1994), 79-96.
 21. A. Harten. High resolution schemes for hyperbolic conservation laws. *Journal of Computational Physics*, **49** (3) (1983), 357-393.
 22. P.K. Sweby. High resolution schemes using flux limiters for hyperbolic conservation laws. *SIAM Journal on Numerical Analysis*, **21** (5) (1984), 995-1011.
 23. W. Hundsdorfer, R.A. Trompert. Method of lines and direct discretization: a comparison for linear advection. *Applied Numerical Mathematics*, **13** (6) (1994), 469-490.
 24. F. Kemm. A comparative study of TVD-limiters—well-known limiters and an introduction of new ones. *International Journal for Numerical Methods in Fluids*, **67** (4) (2011), 404-440.
 25. Y.N. Jeng, U.J. Payne. An adaptive TVD limiter. *Journal of Computational Physics*, **118** (2) (1995), 229-241.
 26. M.K. Kadalbajoo, R. Kumar. A high resolution total variation diminishing scheme for hyperbolic conservation law and related problems. *Applied Mathematics and Computation*, **175** (2) (2006), 1556-1573.
 27. P.L. Roe. Some contributions to the modelling of discontinuous flows. *Large-scale Computations in Fluid Mechanics*, **1** (1985), 163-193.

28. M. Arora, P.L. Roe. A well-behaved TVD limiter for high-resolution calculations of unsteady flow. *Journal of Computational Physics*, **132** (1) (1997), 3-11.
29. G.D. Van Albada, B. Van Leer, W.W. Roberts Jr. A comparative study of computational methods in cosmic gas dynamics. *Astronomy and Astrophysics*, **108** (1982), 76-84.
30. M.A. Alves, P.J. Oliveira, F.T. Pinho. A convergent and universally bounded interpolation scheme for the treatment of advection. *International Journal for Numerical Methods in Fluids*, **41** (1) (2003), 47-75.
31. H.Q. Yang, A.J. Przekwas. A comparative study of advanced shock-capturing schemes applied to Burgers' equation. *Journal of Computational Physics*, **102**(1) (1992), 139-159.
32. B. Van Leer. Towards the ultimate conservative difference scheme. II. Monotonicity and conservation combined in a second-order scheme. *Journal of Computational Physics*, **14** (4) (1974), 361-370.
33. J. Hou, F. Simons, R. Hinkelmann. A new TVD method for advection simulation on 2D unstructured grids. *International Journal for Numerical Methods in Fluids*, **71** (10) (2013), 1260-1281.
34. C.M. Oldenburg, K. Pruess. Simulation of propagating fronts in geothermal reservoirs with the implicit Leonard total variation diminishing scheme. *Geothermics*, **29** (1) (2000), 1-25.
35. M. Čada, M. Torrilhon. Compact third-order limiter functions for finite volume methods. *Journal of Computational Physics*, **228** (11) (2009), 4118-4145.
36. W. Hundsdorfer, B. Koren, M. Vanloon, J.G. Verwer. A positive finite-difference advection scheme. *Journal of Computational Physics*, **117** (1) (1995), 35-46.
37. J. Liu, M. Delshad, G.A. Pope, K. Sepehrnoori. Application of higher-order flux-limited methods in compositional simulation. *Transport in Porous media*, **16** (1) (1994), 1-29.
38. M.S. Darwish, F. Moukalled. TVD schemes for unstructured grids. *International Journal of Heat and Mass Transfer*, **46** (4) (2003), 599-611.
39. V. Przulj, B. Basara. Bounded convection schemes for unstructured grids. *15th AIAA Computational Fluid Dynamics Conference*, Anaheim, CA, (2001).
40. F.S. Lien, M.A. Leschziner. Upstream monotonic interpolation for scalar transport with application to complex turbulent flows. *International Journal for Numerical Methods in Fluids*, **19** (6) (1994), 527-548.
41. B. Song, G.R. Liu, K.Y. Lam, R.S. Amano. On a higher-order bounded discretization scheme. *International Journal for Numerical Methods in Fluids*, **32** (7) (2000), 881-897.
42. K.B. Kuan, C.A. Lin. Adaptive QUICK-based scheme to approximate convective

- transport. *AIAA Journal*, **38** (12) (2000), 2233-2237.
43. H. Jasak, H.G. Weller, A.D. Gosman. High resolution NVD differencing scheme for arbitrarily unstructured meshes. *International Journal for Numerical Methods in Fluids*, **31** (2) (1999), 431-449.
 44. L.X. Li, H.S. Liao, L.J. Qi. An improved r -factor algorithm for TVD schemes. *International Journal of Heat and Mass Transfer*, **51** (3) (2008), 610-617.
 45. J. Hou, F. Simons, R. Hinkelmann. Improved total variation diminishing schemes for advection simulation on arbitrary grids. *International Journal for Numerical Methods in Fluids*, **70** (3) (2012), 359-382.
 46. S. Bidadi, S.L. Rani. Quantification of numerical diffusivity due to TVD schemes in the advection equation. *Journal of Computational Physics*, **261** (2014), 65-82.
 47. B. Van Leer. Upwind-difference methods for aerodynamic problems governed by the Euler equations. *In Large-scale Computations in Fluid Mechanics*, **1** (1985), 327-336.
 48. B.P. Leonard. The ULTIMATE conservative difference scheme applied to unsteady one-dimensional advection. *Computer Methods in Applied Mechanics and Engineering*, **88** (1) (1991), 17-74.
 49. P.H. Gaskell, A.K.C. Lau. Curvature-compensated convective transport: SMART, A new boundedness-preserving transport algorithm. *International Journal for Numerical Methods in Fluids*, **8** (6) (1988), 617-641.
 50. C. Bruner, R. Walters, Parallelization of the Euler equations on unstructured grids, *AIAA Paper*, (97-1894) (1997), 446-470.

Table 1 Accuracy of various r -factor algorithms for ten classical SS-TVD flux-limiters on arbitrary unstructured grids in Test 1, 2, 3 and 4.

Scheme	Step		Sine-Square		Semi-Ellipse		Double-Step Rotation	
	Darwish's	FFISAM	Darwish's	FFISAM	Darwish's	FFISAM	Darwish's	FFISAM
Superbee	6.88E-03	6.41E-03	3.02E-03	2.30E-03	3.48E-03	3.62E-03	1.04E-02	1.01E-02
Koren	8.10E-03	7.70E-03	3.33E-03	3.00E-03	4.37E-03	4.15E-03	1.18E-02	1.16E-02
MUSCL	8.17E-03	7.85E-03	3.26E-03	3.04E-03	4.39E-03	4.21E-03	1.18E-02	1.16E-02
WACEB	8.07E-03	7.66E-03	3.42E-03	3.04E-03	4.38E-03	4.14E-03	1.19E-02	1.17E-02
TCDF	8.28E-03	7.89E-03	3.93E-03	3.52E-03	4.59E-03	4.22E-03	1.22E-02	1.19E-02
UMIST	9.00E-03	8.70E-03	5.17E-03	4.84E-03	5.40E-03	5.07E-03	1.28E-02	1.26E-02
Minmod	9.77E-03	9.50E-03	7.24E-03	6.87E-03	6.63E-03	6.16E-03	1.41E-02	1.39E-02
OSPRE	8.87E-03	8.53E-03	5.26E-03	4.72E-03	5.15E-03	4.72E-03	1.28E-02	1.24E-02
Albada	9.04E-03	8.70E-03	5.63E-03	5.13E-03	5.48E-03	4.95E-03	1.31E-02	1.28E-02
Harmonic	8.63E-03	8.32E-03	4.43E-03	4.02E-03	4.90E-03	4.55E-03	1.24E-02	1.22E-02

Table 2 Accuracy of various r -factor algorithms for ten classical SS-TVD flux-limiters on an unstructured triangular grid in Test 4.

Algorithm	Superbee	Koren	MUSCL	WACEB	TCDF	UMIST	Minmod	OSPRE	Albada	Harmonic
Darwish's	1.11E-02	1.34E-02	1.37E-02	1.33E-02	1.36E-02	1.48E-02	1.62E-02	1.46E-02	1.48E-02	1.43E-02
Hou's	1.08E-02	1.33E-02	1.36E-02	1.32E-02	1.34E-02	1.47E-02	1.60E-02	1.42E-02	1.45E-02	1.41E-02
FFISAM	1.01E-02	1.28E-02	1.31E-02	1.27E-02	1.30E-02	1.42E-02	1.57E-02	1.39E-02	1.42E-02	1.36E-02

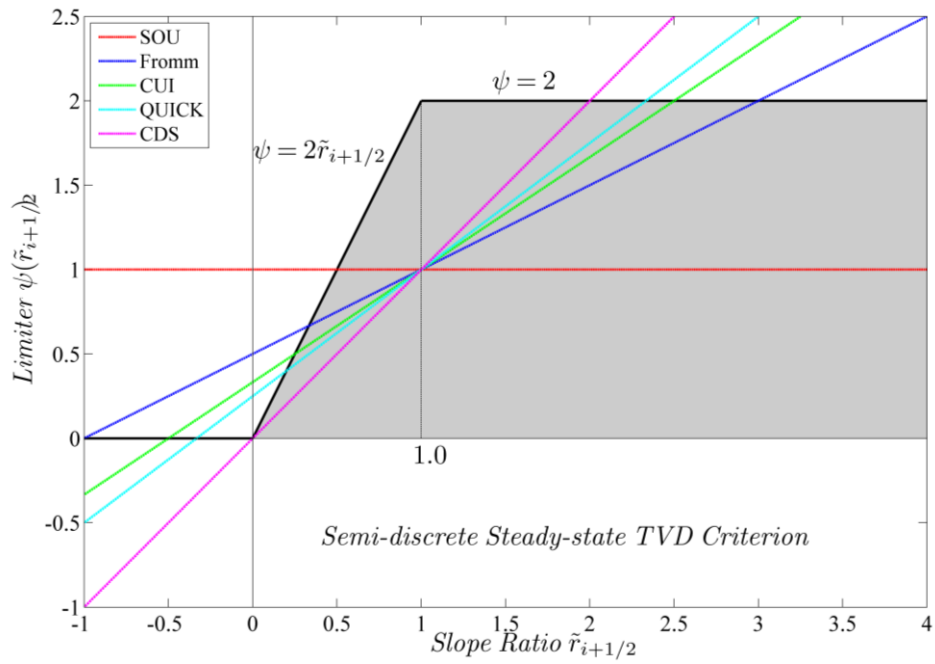


Figure 1. A graphical representation of the SS-TVD criterion (viz. Sweby's TVD criterion) and the k -schemes.

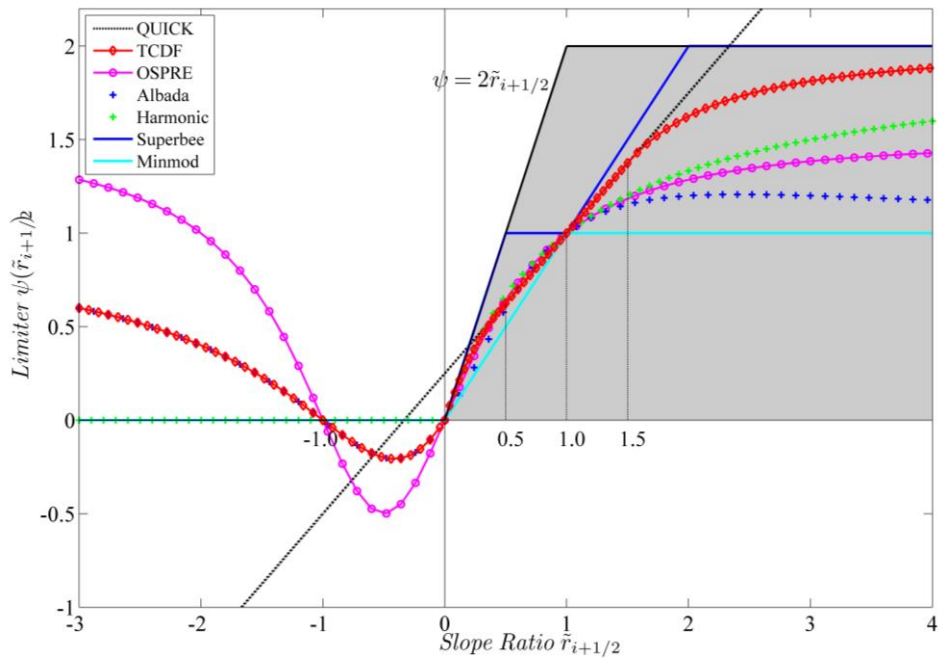


Figure 2. A graphical representation of several existing classical SS-TVD flux-limiters.

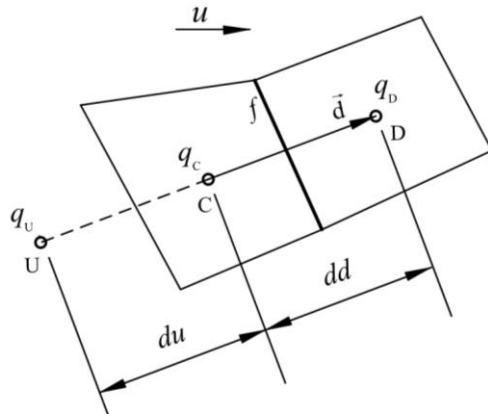


Figure 3. The more appropriate notation adopted for arbitrary unstructured grids and a graphical representation of Darwish's r -factor algorithm.

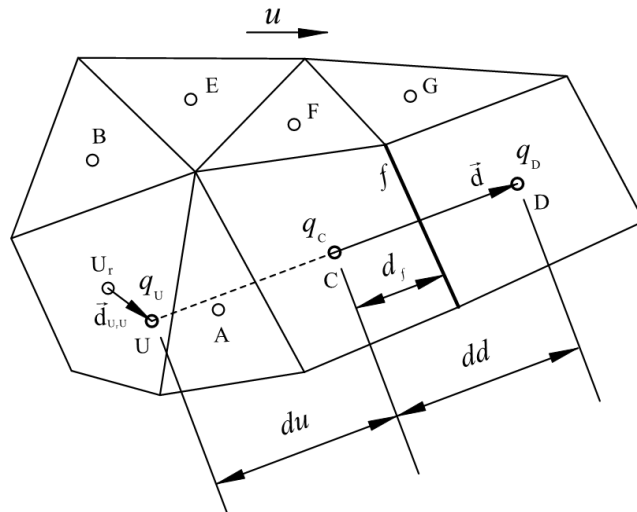


Figure 4. A graphical representation of Li's r -factor algorithm on two-dimensional arbitrary unstructured grids.

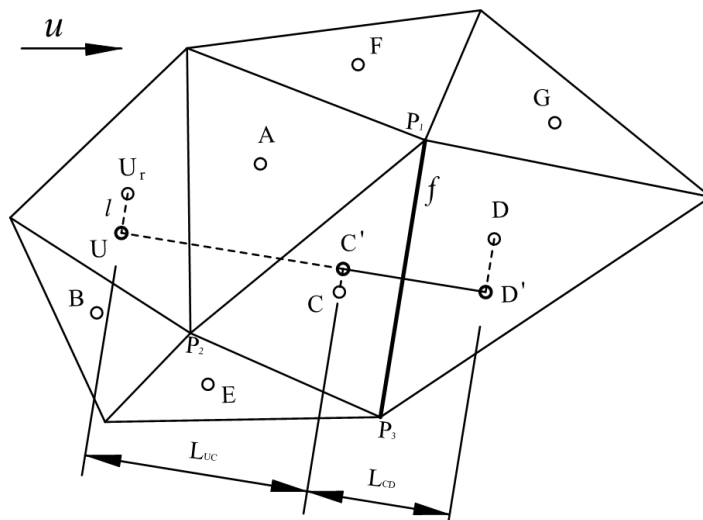
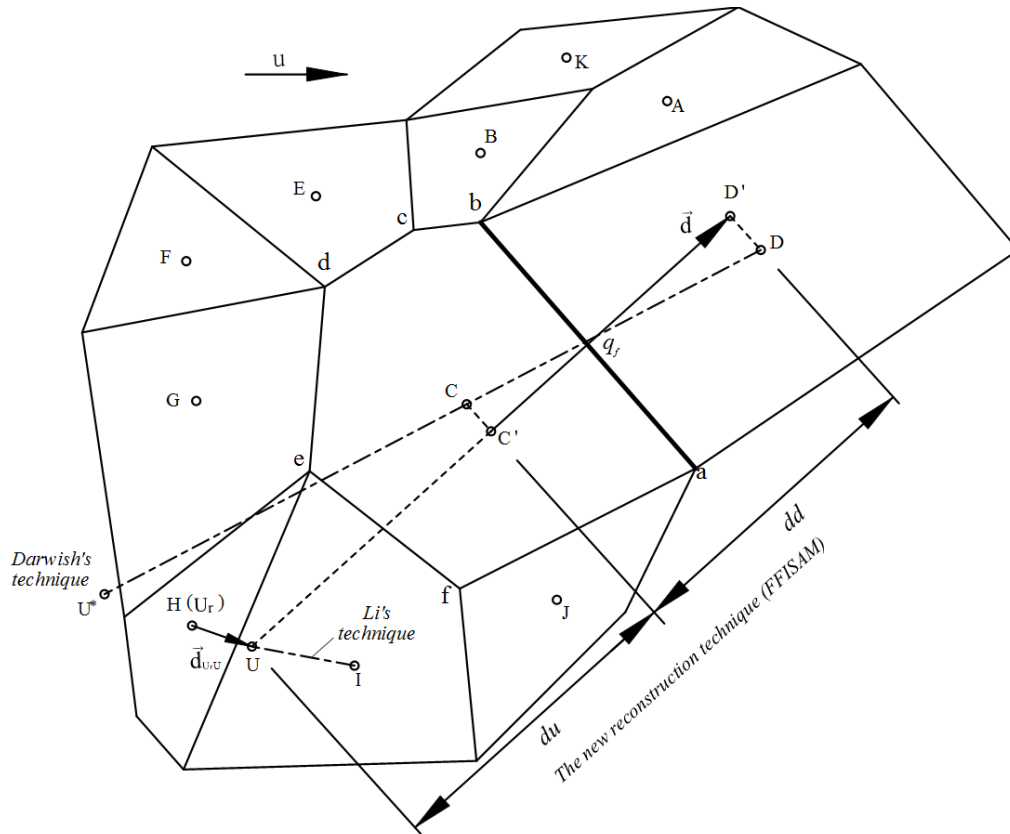
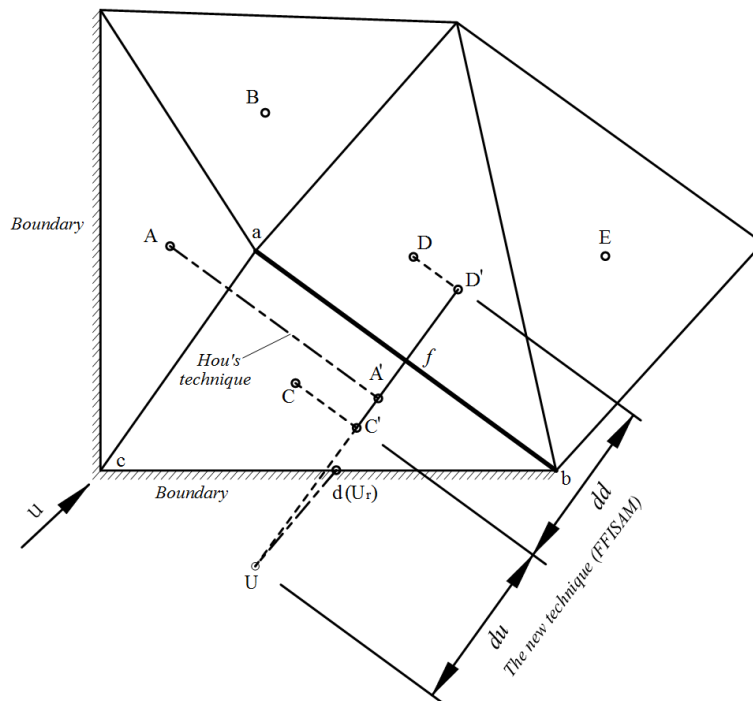


Figure 5. A graphical representation of Hou's r -factor algorithm on two-dimensional unstructured triangular grids.



Arbitrary Unstructured Grid Arrangement

Figure 6. A graphical representation of the newly-proposed r -factor algorithm (FFISAM) on arbitrary unstructured grids (for the inner domain).



Arbitrary Unstructured Grid Arrangement

Figure 7. A graphical representation of the newly-proposed r -factor algorithm (FFISAM) on arbitrary unstructured grids (at the boundary).

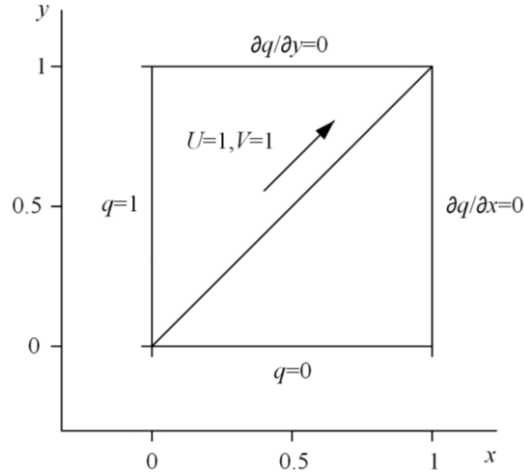


Figure 8. Pure convection of a step profile by a uniform velocity field.

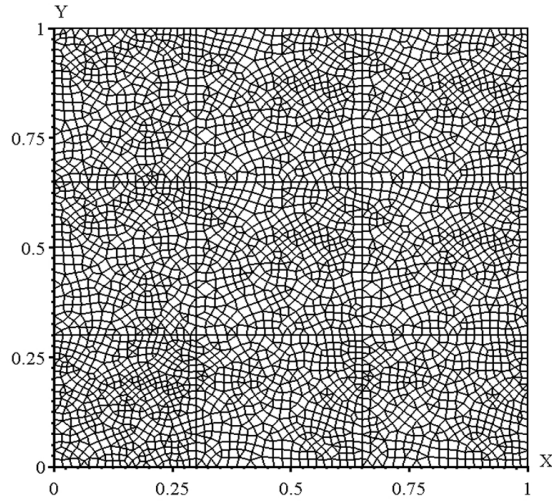


Figure 9. Physical domain and arbitrary unstructured mesh used for the pure convection of a step profile.

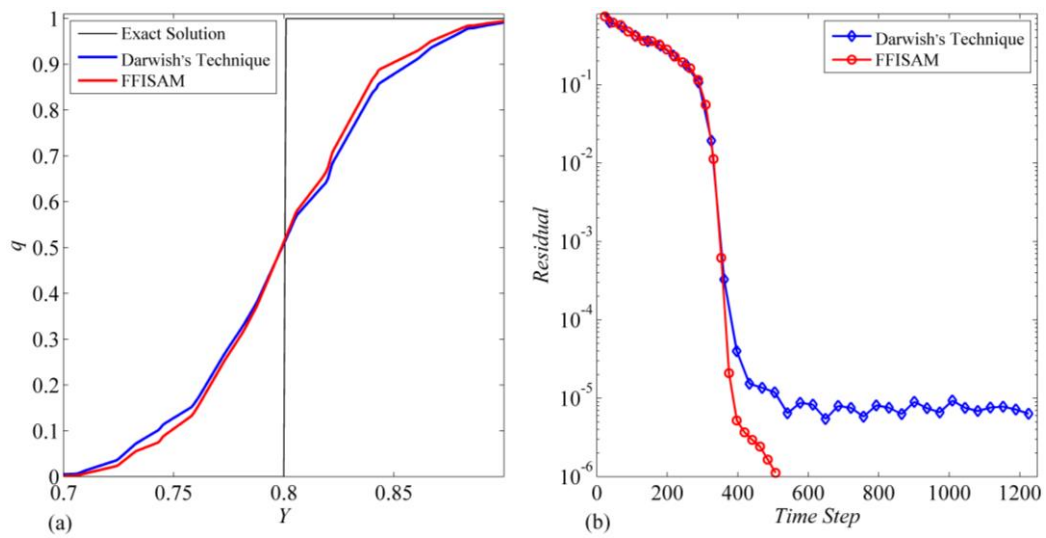


Figure 10. Comparison of accuracy and convergence for various r -factor algorithms ($Cu=1.0$) in Test 1 (Koren / Step): (a) scalar profiles; (b) decay of the norm of the residuals.

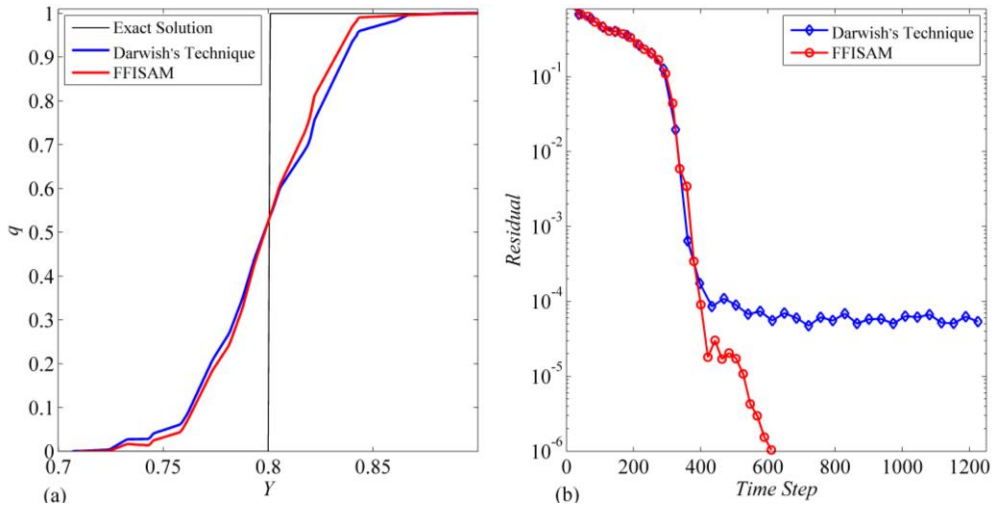


Figure 11. Comparison of accuracy and convergence for various r -factor algorithms ($Cu=1.0$) in Test 1 (Superbee / Step): (a) scalar profiles; (b) decay of the norm of the residuals.

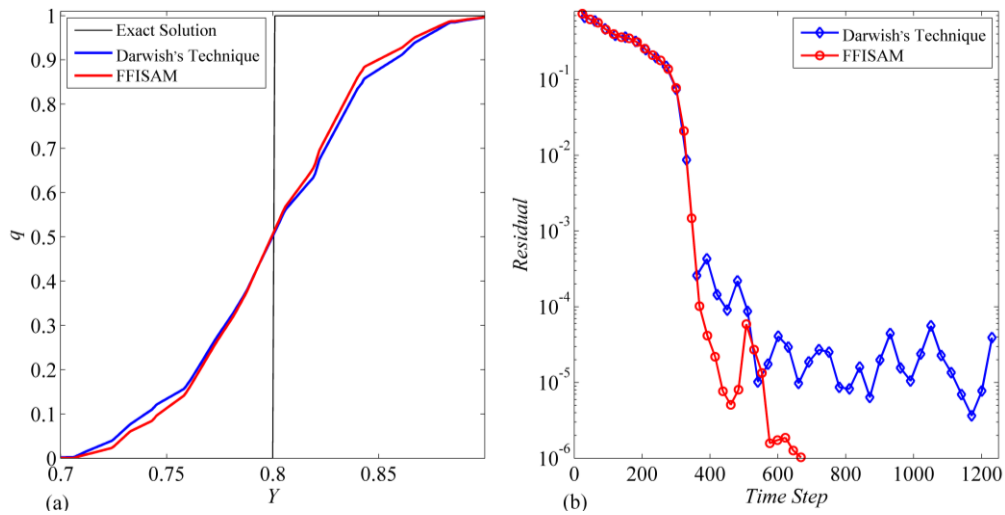


Figure 12. Comparison of accuracy and convergence for various r -factor algorithms ($Cu=1.0$) in Test 1 (MUSCL / Step): (a) scalar profiles; (b) decay of the norm of the residuals.

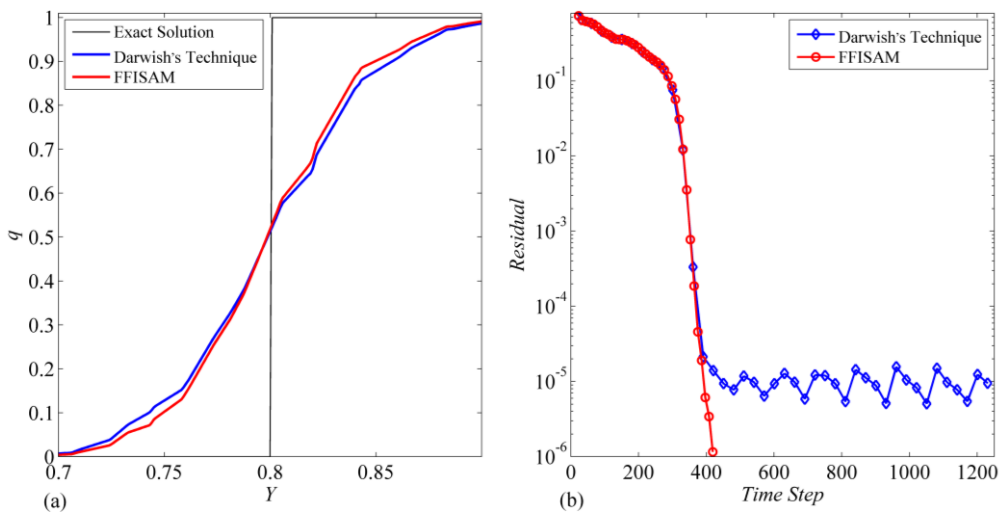


Figure 13. Comparison of accuracy and convergence for various r -factor algorithms ($Cu=1.0$) in Test 1 (WACEB / Step): (a) scalar profiles; (b) decay of the norm of the residuals.

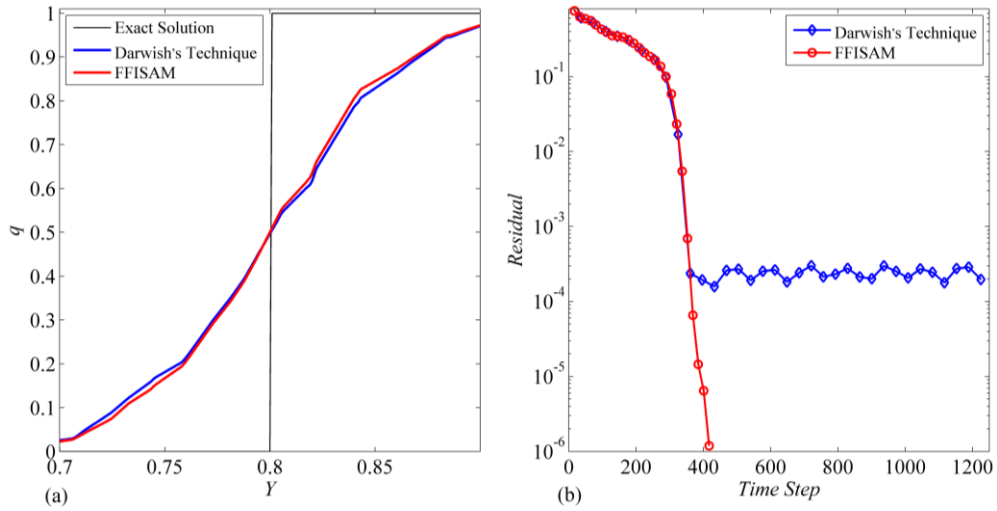


Figure 14. Comparison of accuracy and convergence for various r -factor algorithms ($Cu=1.0$) in Test 1 (UMIST / Step): (a) scalar profiles; (b) decay of the norm of the residuals.

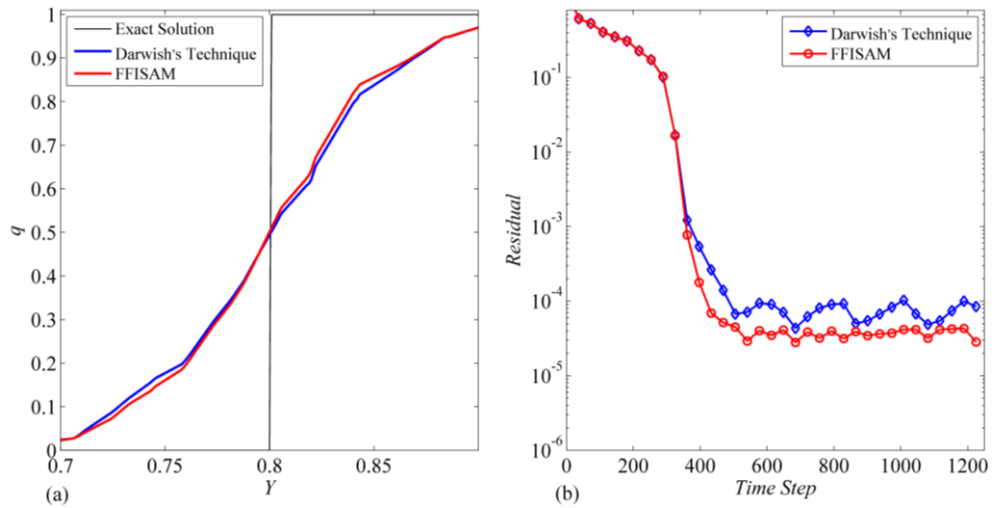


Figure 15. Comparison of accuracy and convergence for various r -factor algorithms ($Cu=1.0$) in Test 1 (OSPRE / Step): (a) scalar profiles; (b) decay of the norm of the residuals.

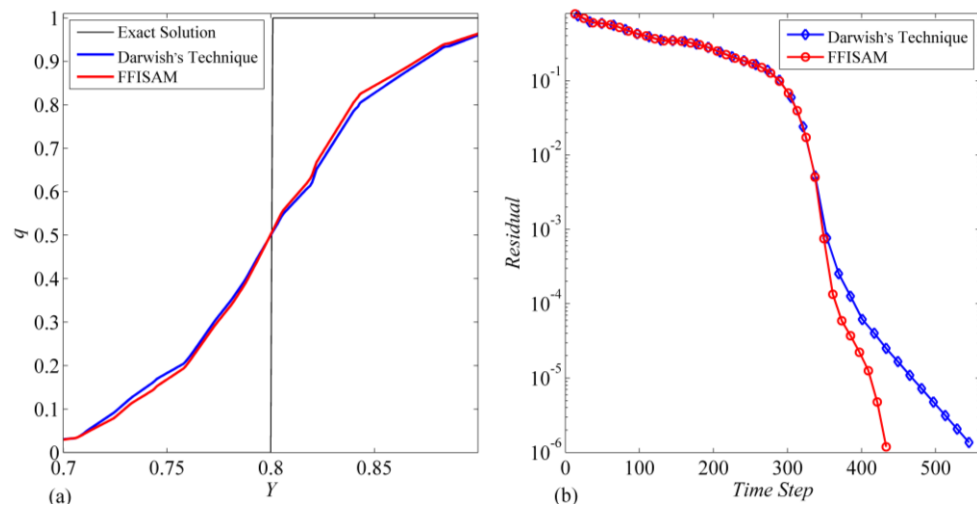


Figure 16. Comparison of accuracy and convergence for various r -factor algorithms ($Cu=1.0$) in Test 1 (Albada / Step): (a) scalar profiles; (b) decay of the norm of the residuals.

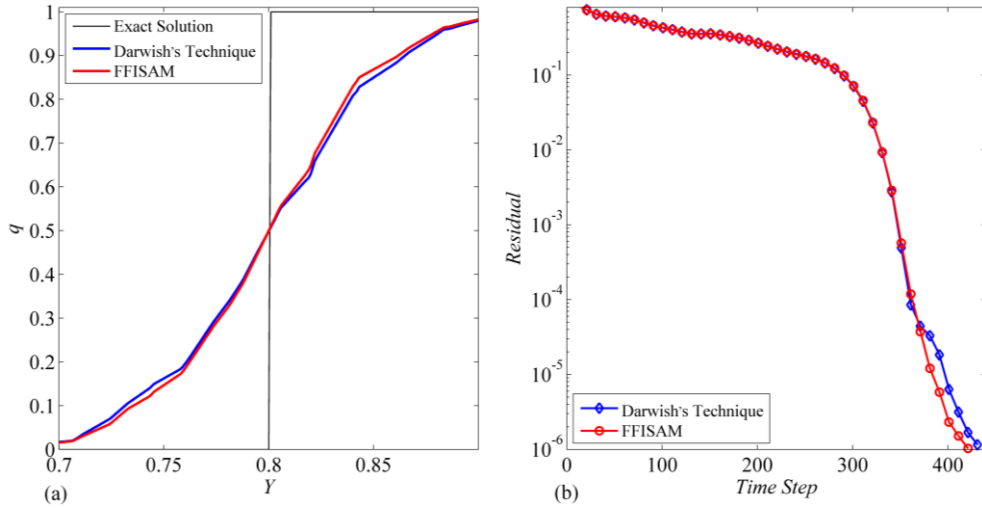


Figure 17. Comparison of accuracy and convergence for various r -factor algorithms ($Cu=1.0$) in Test 1 (Harmonic / Step): (a) scalar profiles; (b) decay of the norm of the residuals.

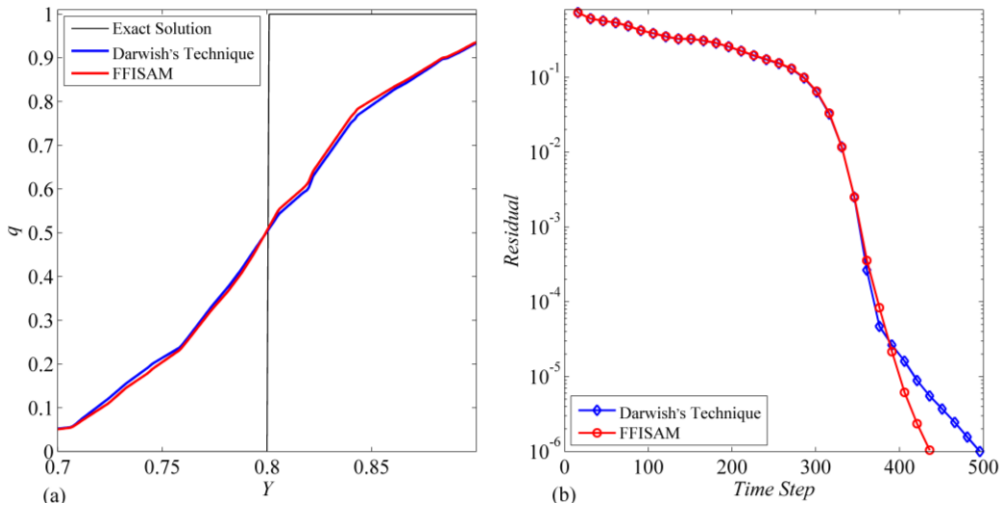


Figure 18. Comparison of accuracy and convergence for various r -factor algorithms ($Cu=1.0$) in Test 1 (Minmod / Step): (a) scalar profiles; (b) decay of the norm of the residuals.

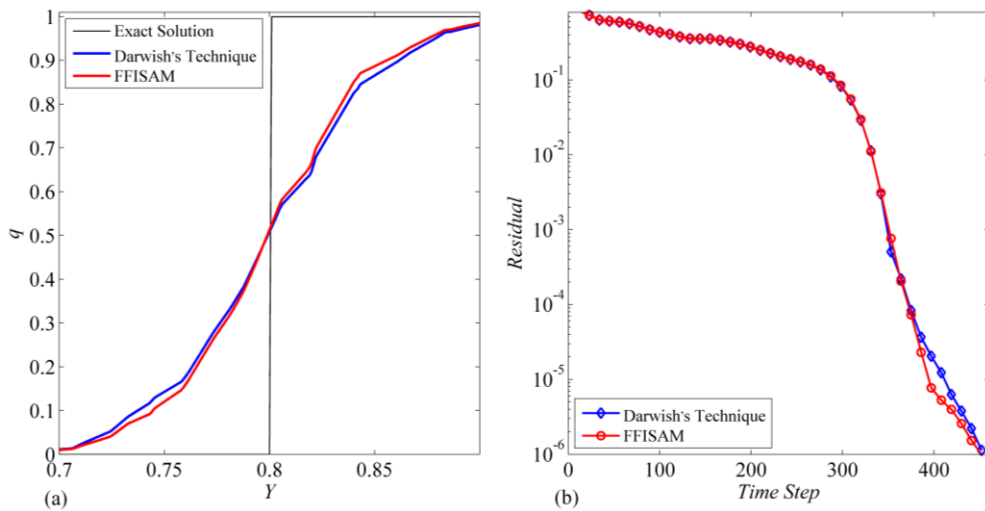


Figure 19. Comparison of accuracy and convergence for various r -factor algorithms ($Cu=1.0$) in Test 1 (TCDF / Step): (a) scalar profiles; (b) decay of the norm of the residuals.

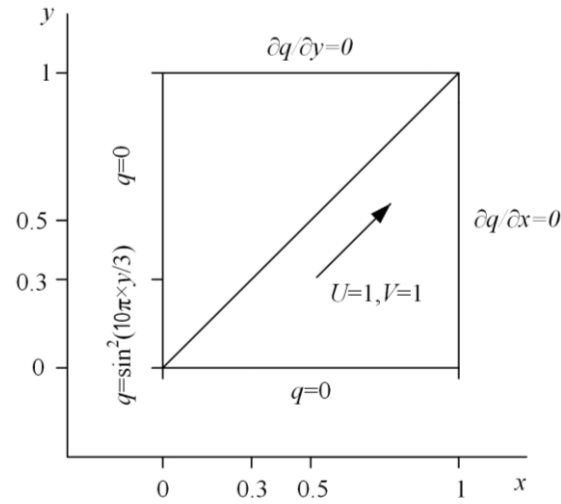


Figure 20. Pure convection of a sine-square profile by a uniform velocity field.

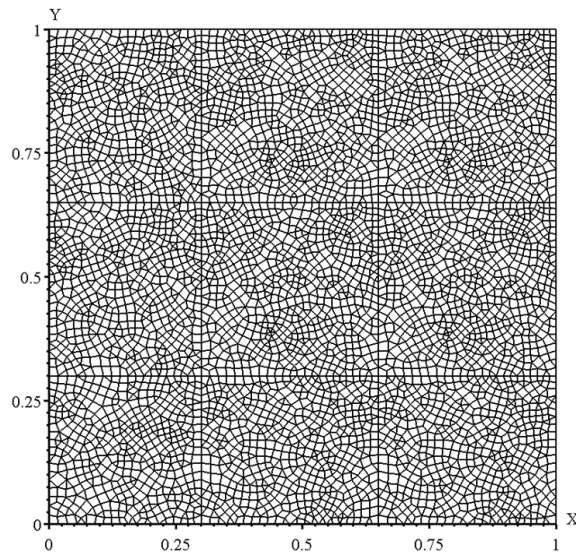


Figure 21. Physical domain and arbitrary unstructured mesh used for the pure convection of a sine-square profile.

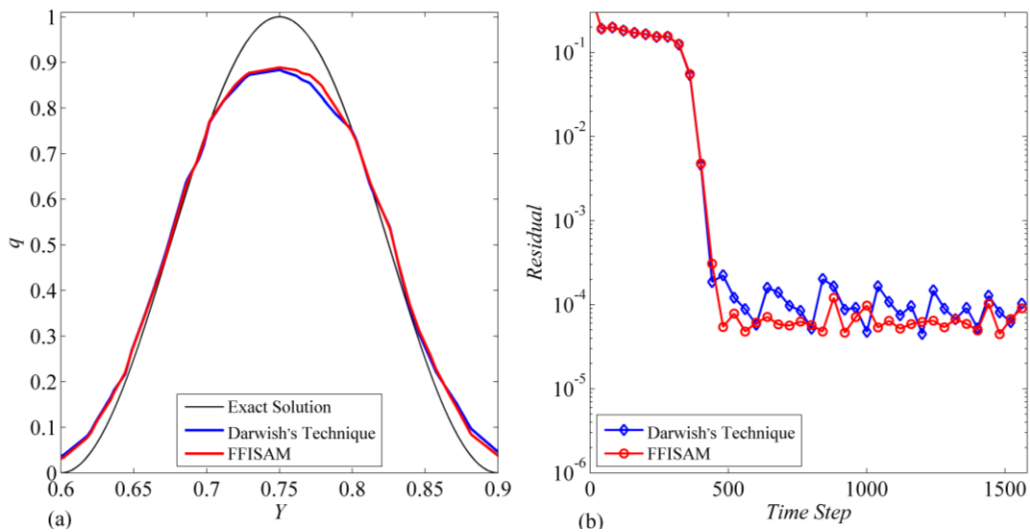


Figure 22. Comparison of accuracy and convergence for various r -factor algorithms ($Cu=1.0$) in Test 2 (Koren / Sine-square): (a) scalar profiles; (b) decay of the norm of the residuals.

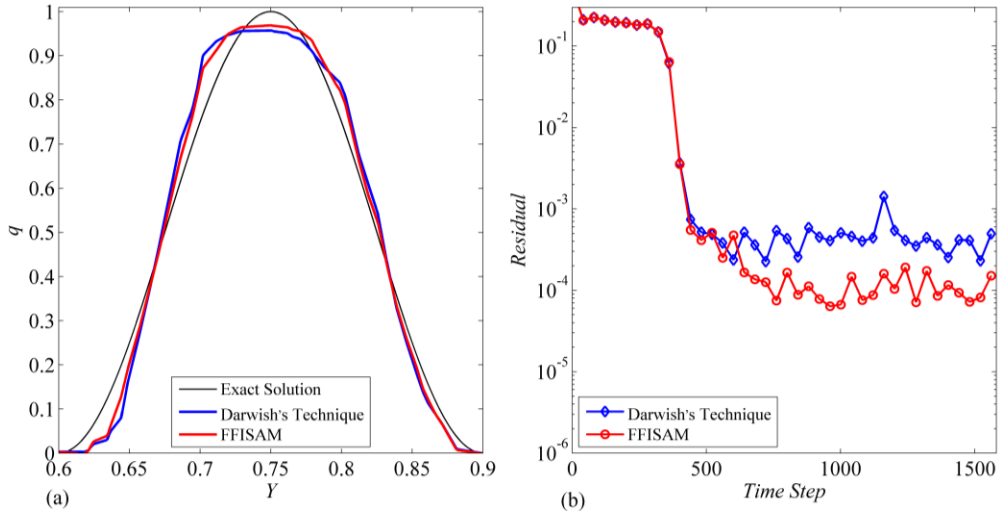


Figure 23. Comparison of accuracy and convergence for various r -factor algorithms ($Cu=1.0$) in Test 2 (Superbee / Sine-square): (a) scalar profiles; (b) decay of the norm of the residuals.

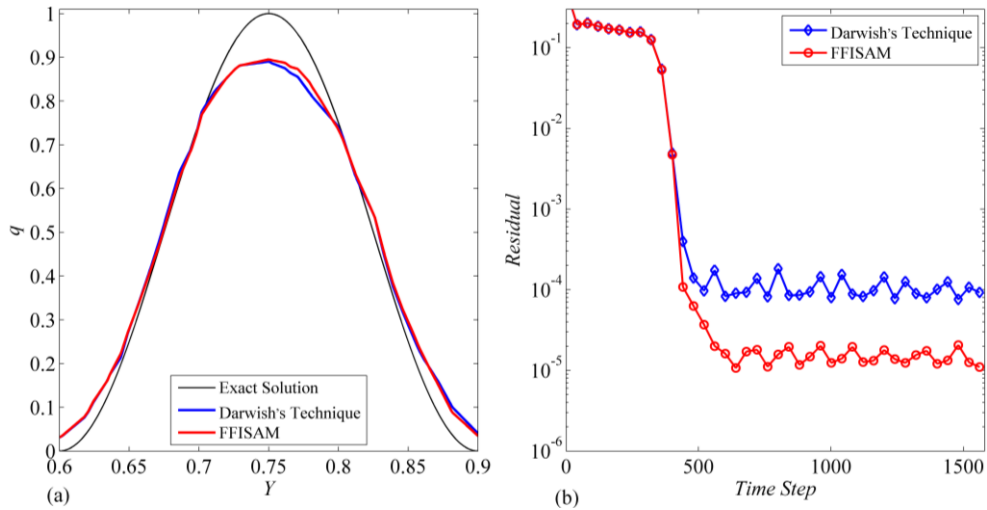


Figure 24. Comparison of accuracy and convergence for various r -factor algorithms ($Cu=1.0$) in Test 2 (MUSCL / Sine-square): (a) scalar profiles; (b) decay of the norm of the residuals.

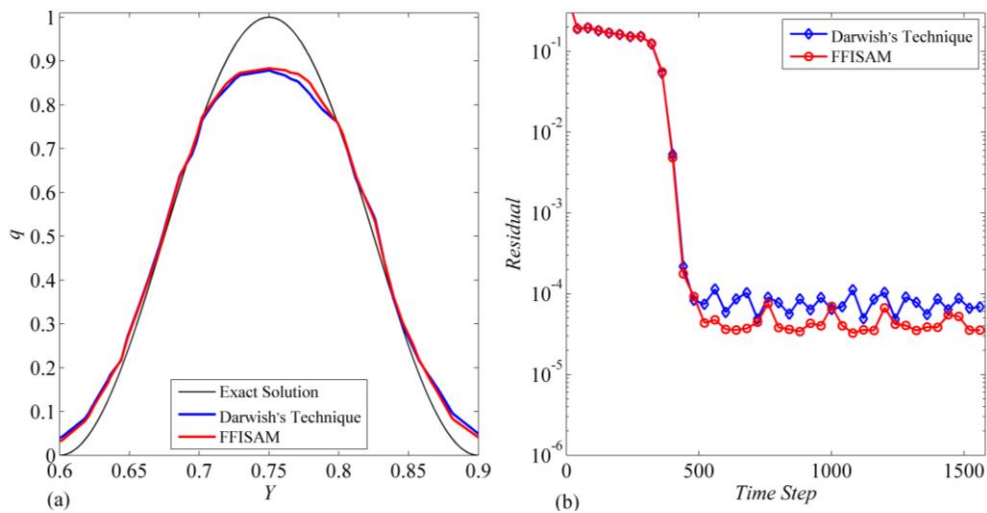


Figure 25. Comparison of accuracy and convergence for various r -factor algorithms ($Cu=1.0$) in Test 2 (WACEB / Sine-square): (a) scalar profiles; (b) decay of the norm of the residuals.

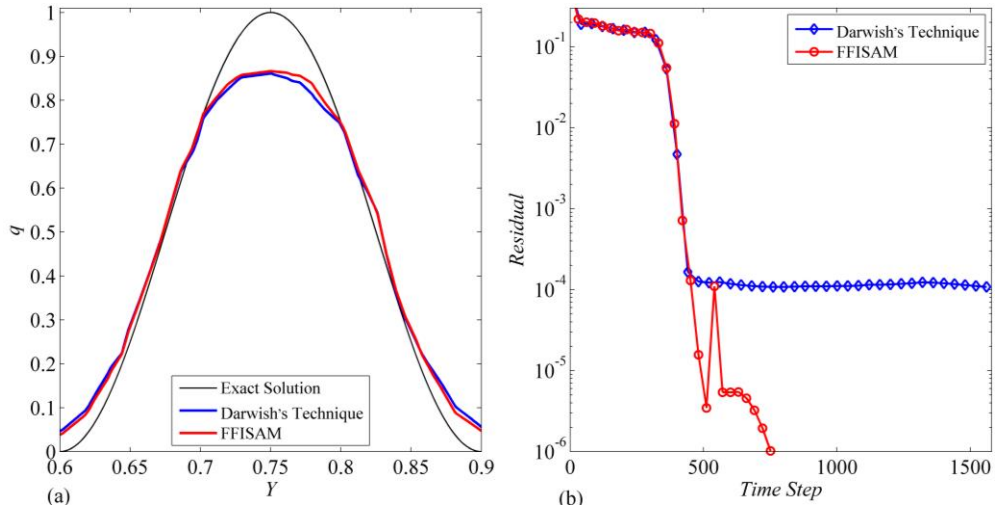


Figure 26. Comparison of accuracy and convergence for various r -factor algorithms ($Cu=1.0$) in Test 2 (TCDF / Sine-square): (a) scalar profiles; (b) decay of the norm of the residuals.

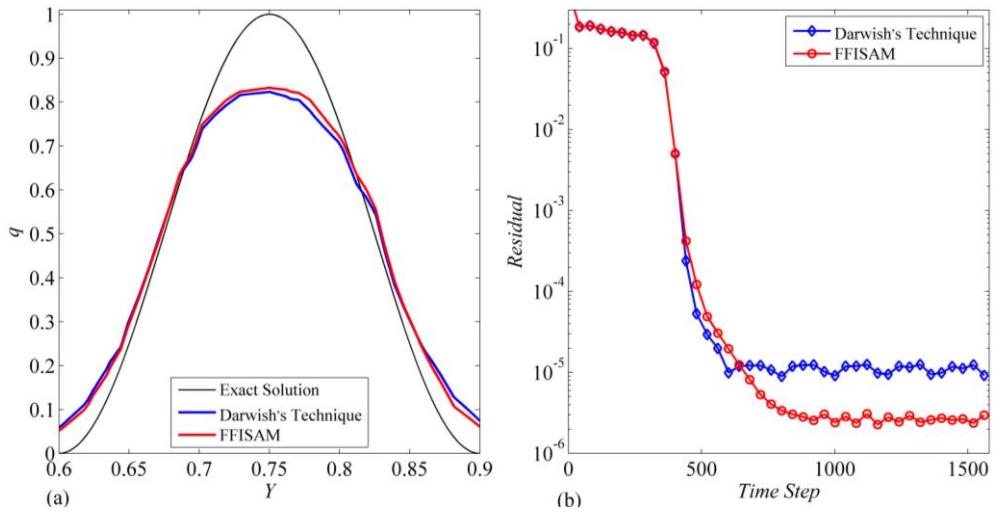


Figure 27. Comparison of accuracy and convergence for various r -factor algorithms ($Cu=1.0$) in Test 2 (OSPRE / Sine-square): (a) scalar profiles; (b) decay of the norm of the residuals.

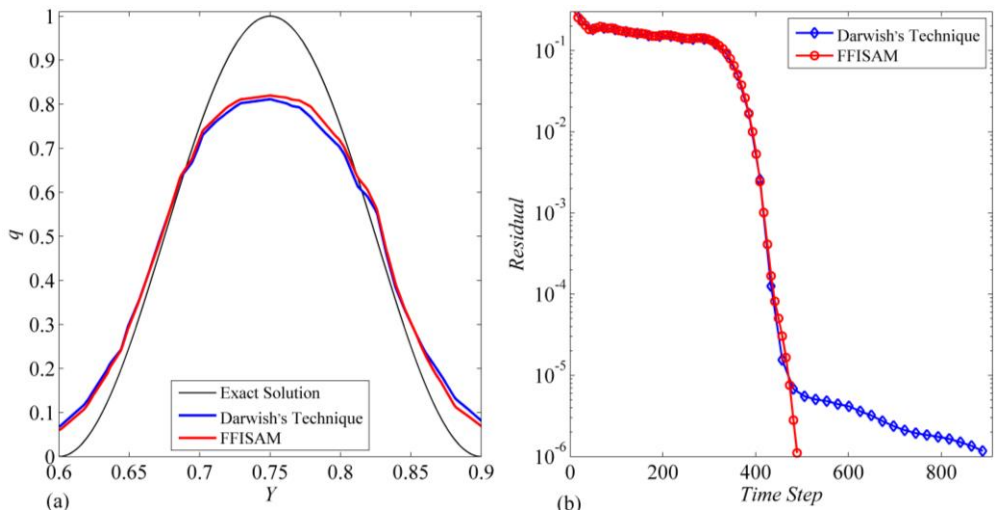


Figure 28. Comparison of accuracy and convergence for various r -factor algorithms ($Cu=1.0$) in Test 2 (Albada / Sine-square): (a) scalar profiles; (b) decay of the norm of the residuals.

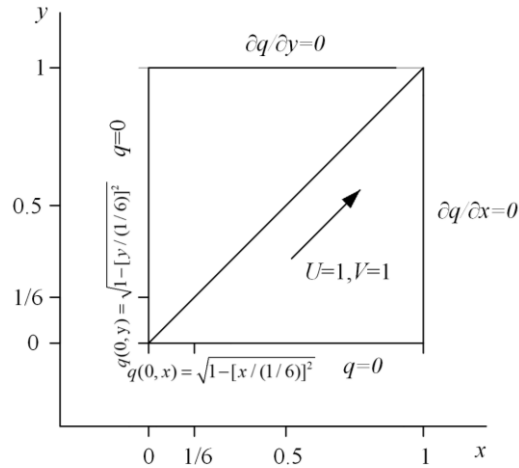


Figure 29. Pure convection of a semi-ellipse profile by a uniform velocity field.

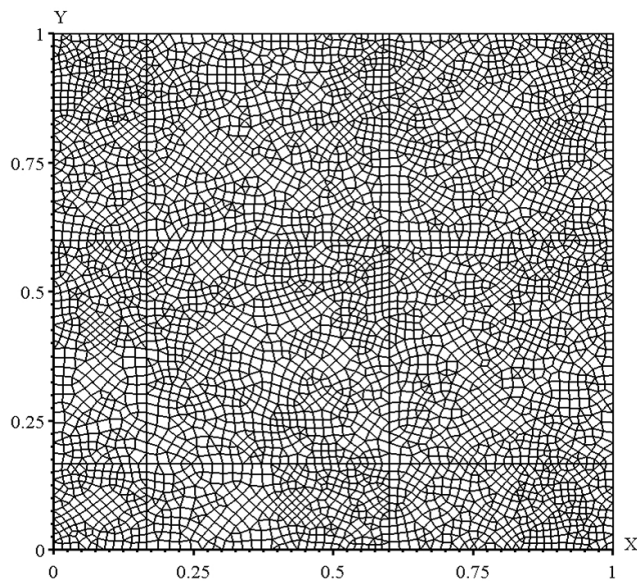


Figure 30. Physical domain and arbitrary unstructured mesh used for the pure convection of a semi-ellipse profile.

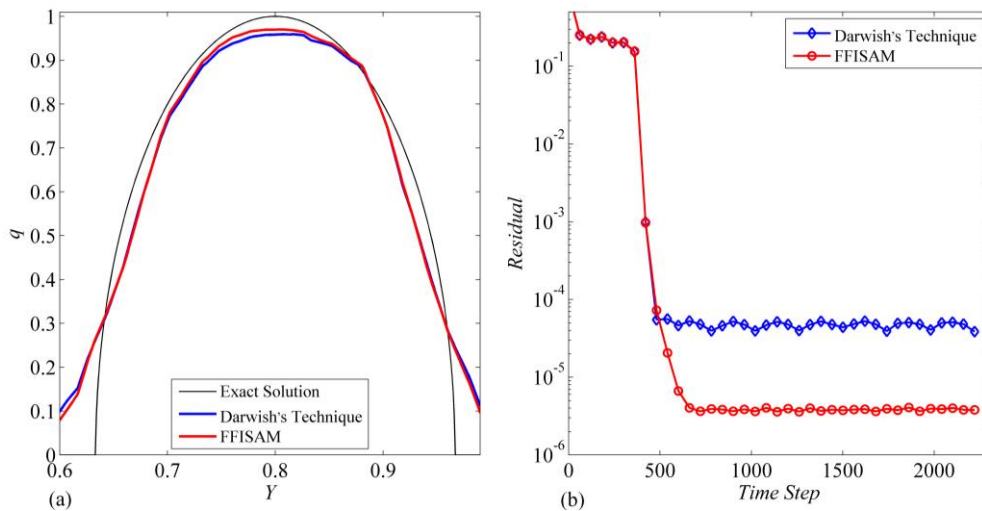


Figure 31. Comparison of accuracy and convergence for various r -factor algorithms ($Cu=1.0$) in Test 3 (Koren / Semi-ellipse): (a) scalar profiles; (b) decay of the norm of the residuals.

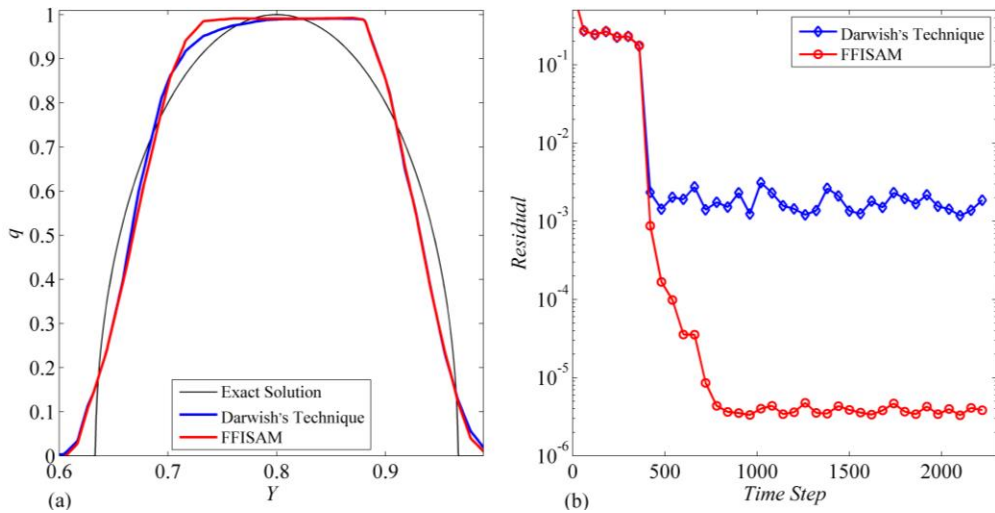


Figure 32. Comparison of accuracy and convergence for various r -factor algorithms ($Cu=1.0$) in Test 3 (Superbee / Semi-ellipse): (a) scalar profiles; (b) decay of the norm of the residuals.

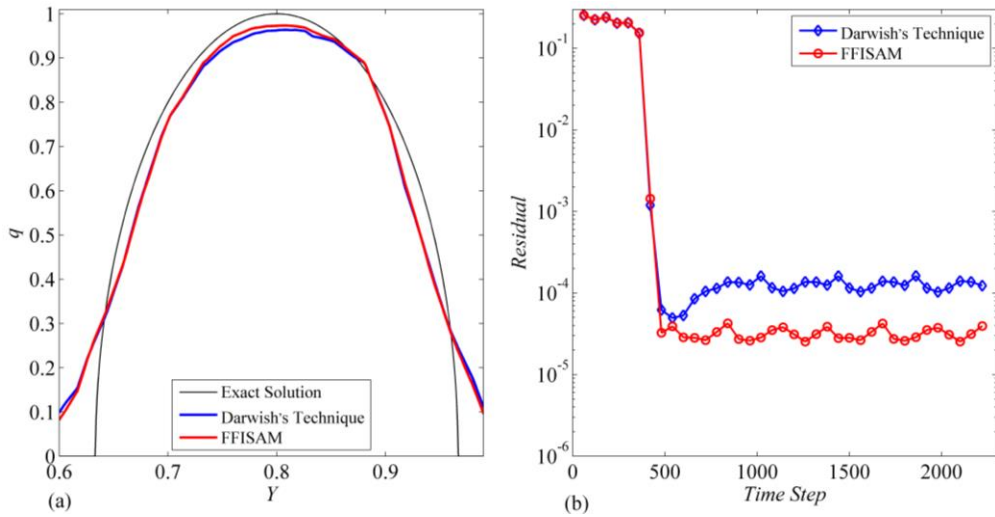


Figure 33. Comparison of accuracy and convergence for various r -factor algorithms ($Cu=1.0$) in Test 3 (MUSCL / Semi-ellipse): (a) scalar profiles; (b) decay of the norm of the residuals.

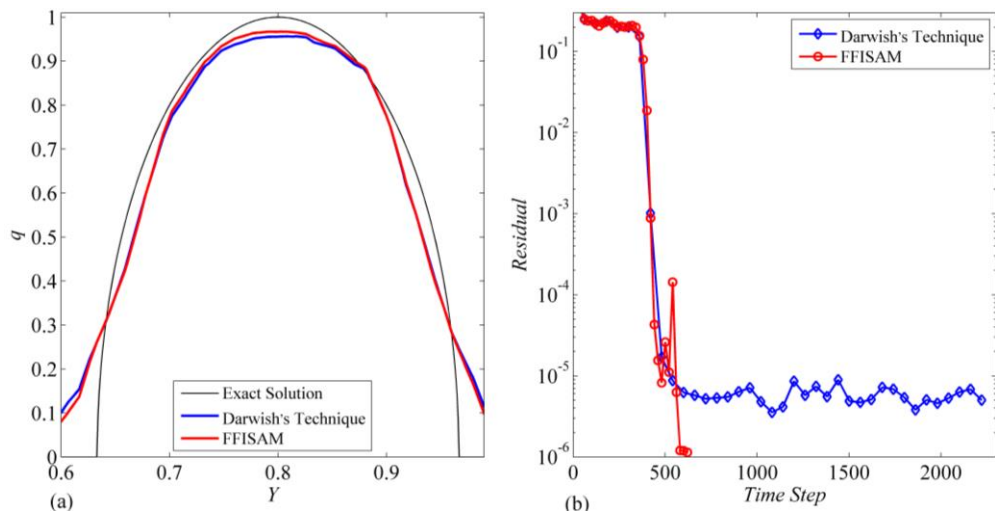


Figure 34. Comparison of accuracy and convergence for various r -factor algorithms ($Cu=1.0$) in Test 3 (WACEB / Semi-ellipse): (a) scalar profiles; (b) decay of the norm of the residuals.

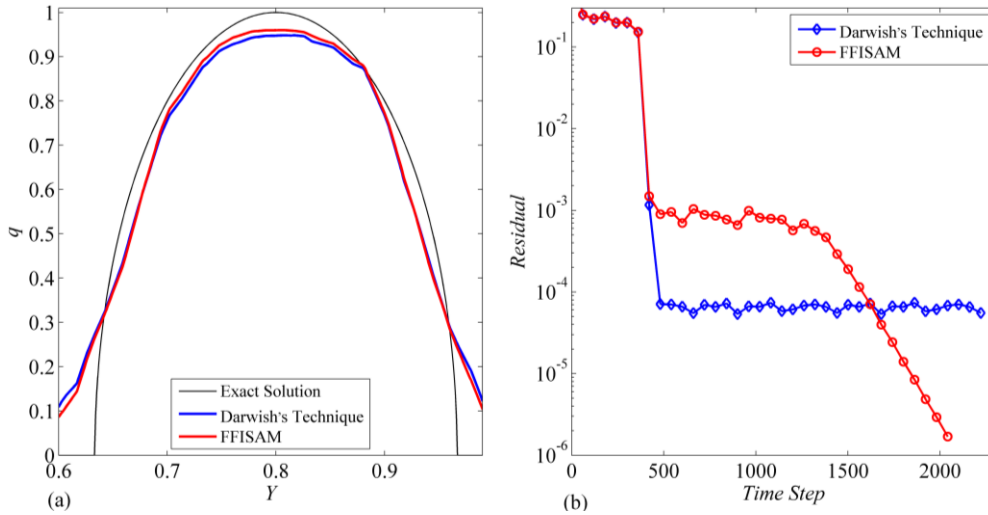


Figure 35. Comparison of accuracy and convergence for various r -factor algorithms ($Cu=1.0$) in Test 3 (TCDF / Semi-ellipse): (a) scalar profiles; (b) decay of the norm of the residuals.

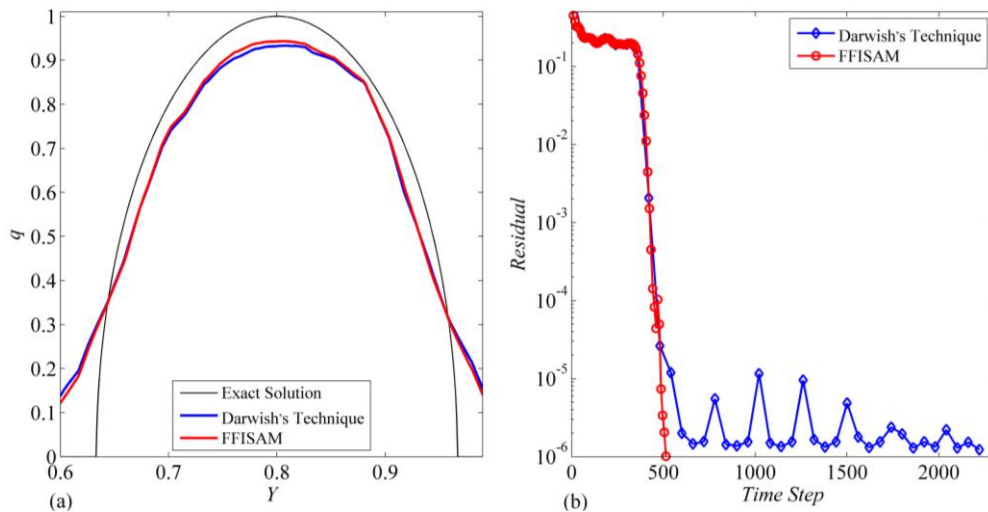


Figure 36. Comparison of accuracy and convergence for various r -factor algorithms ($Cu=1.0$) in Test 3 (UMIST / Semi-ellipse): (a) scalar profiles; (b) decay of the norm of the residuals.

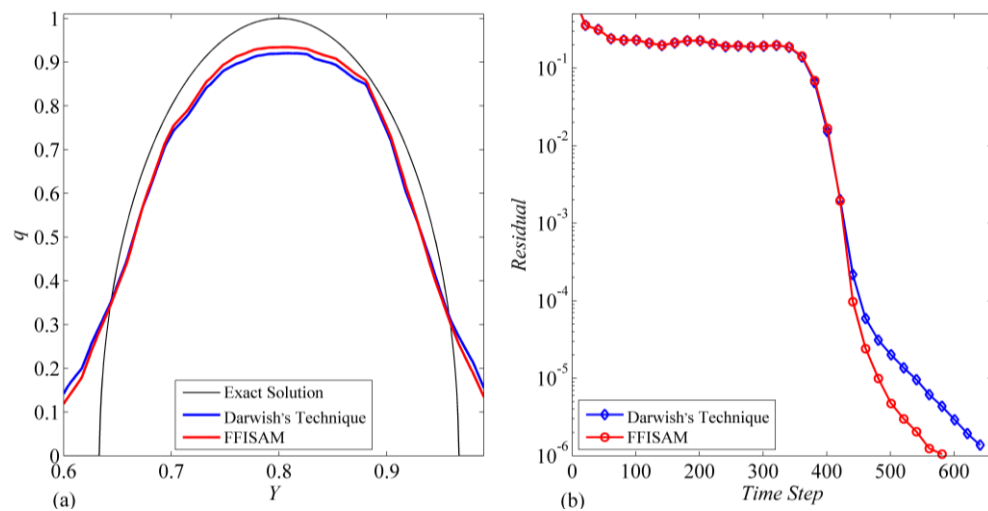


Figure 37. Comparison of accuracy and convergence for various r -factor algorithms ($Cu=1.0$) in Test 3 (Albada / Semi-ellipse): (a) scalar profiles; (b) decay of the norm of the residuals.

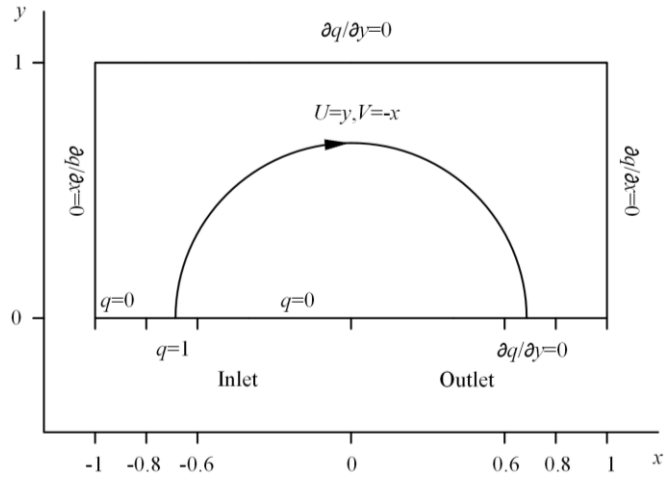


Figure 38. Pure convection of a double-step profile in a rotational velocity field.

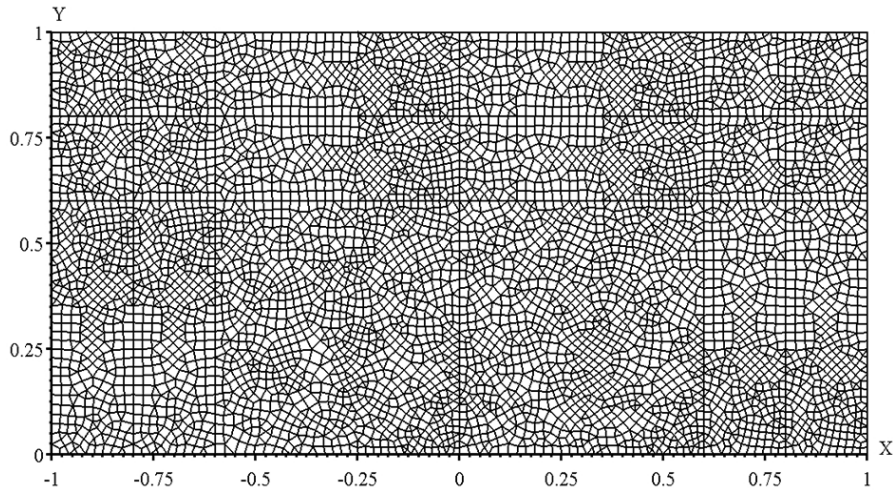


Figure 39. Physical domain and arbitrary unstructured mesh used for the pure convection of a double-step profile in a rotational velocity field.

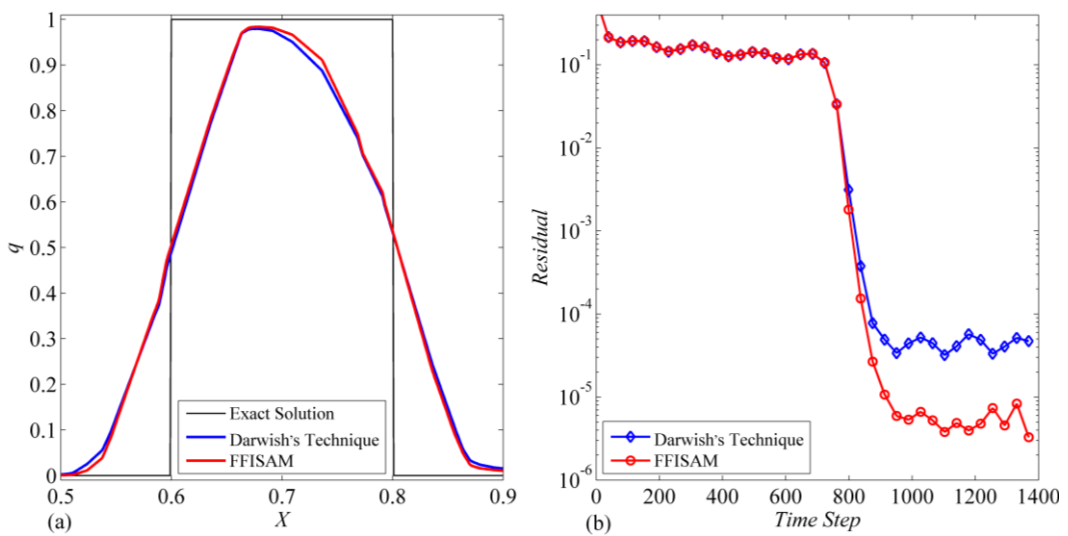


Figure 40. Comparison of accuracy and convergence for various r -factor algorithms ($Cu=1.0$) in Test 4 (Koren / Double-step Rotation / Arbitrary unstructured grid): (a) scalar profiles; (b) decay of the norm of the residuals.

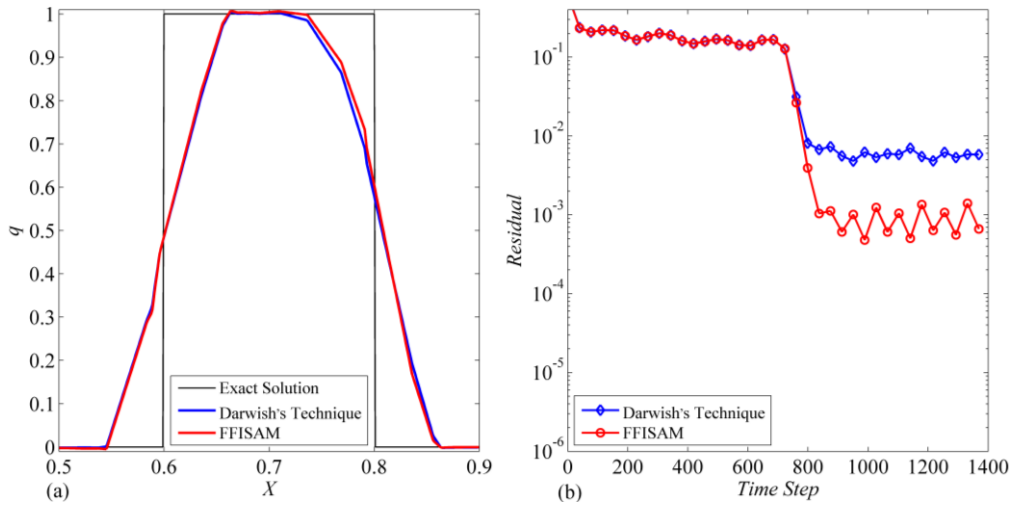


Figure 41. Comparison of accuracy and convergence for various r -factor algorithms ($Cu=1.0$) in Test 4 (Superbee / Double-step Rotation / Arbitrary unstructured grid): (a) scalar profiles; (b) decay of the norm of the residuals.

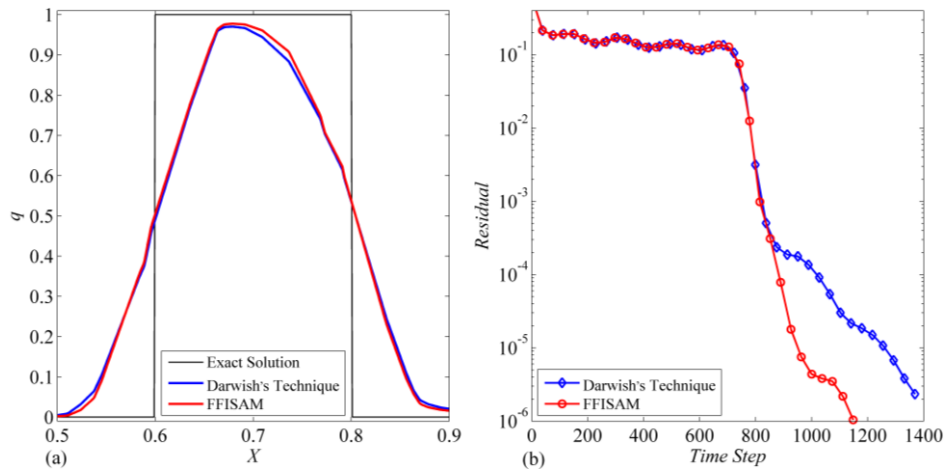


Figure 42. Comparison of accuracy and convergence for various r -factor algorithms ($Cu=1.0$) in Test 4 (WACEB / Double-step Rotation / Arbitrary unstructured grid): (a) scalar profiles; (b) decay of the norm of the residuals.

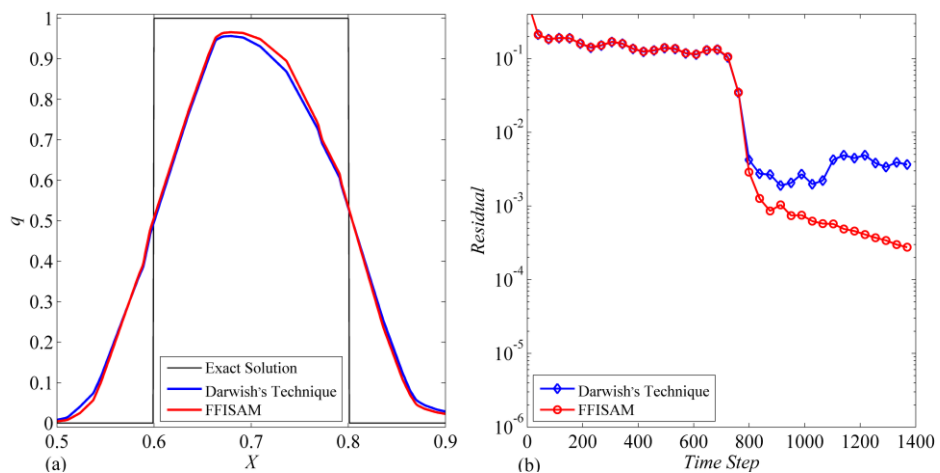


Figure 43. Comparison of accuracy and convergence for various r -factor algorithms ($Cu=1.0$) in Test 4 (TCDF / Double-step Rotation / Arbitrary unstructured grid): (a) scalar profiles; (b) decay of the norm of the residuals.

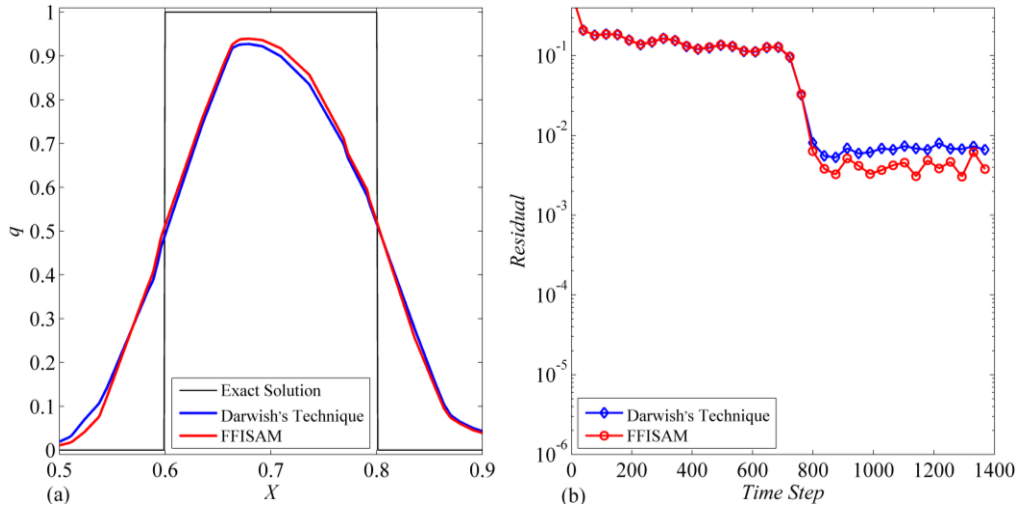


Figure 44. Comparison of accuracy and convergence for various r -factor algorithms ($Cu=1.0$) in Test 4 (OSPRES / Double-step Rotation / Arbitrary unstructured grid): (a) scalar profiles; (b) decay of the norm of the residuals.

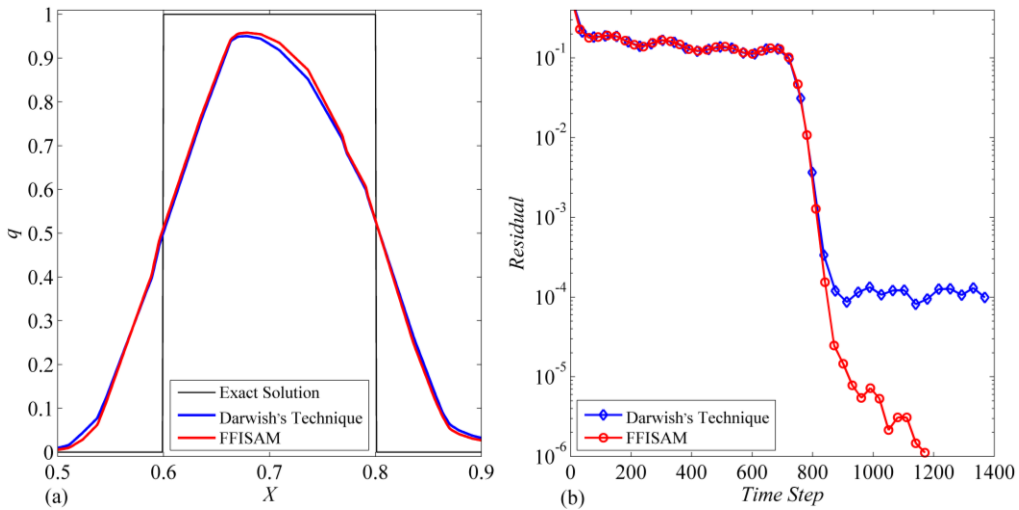


Figure 45. Comparison of accuracy and convergence for various r -factor algorithms ($Cu=1.0$) in Test 4 (Harmonic / Double-step Rotation / Arbitrary unstructured grid): (a) scalar profiles; (b) decay of the norm of the residuals.

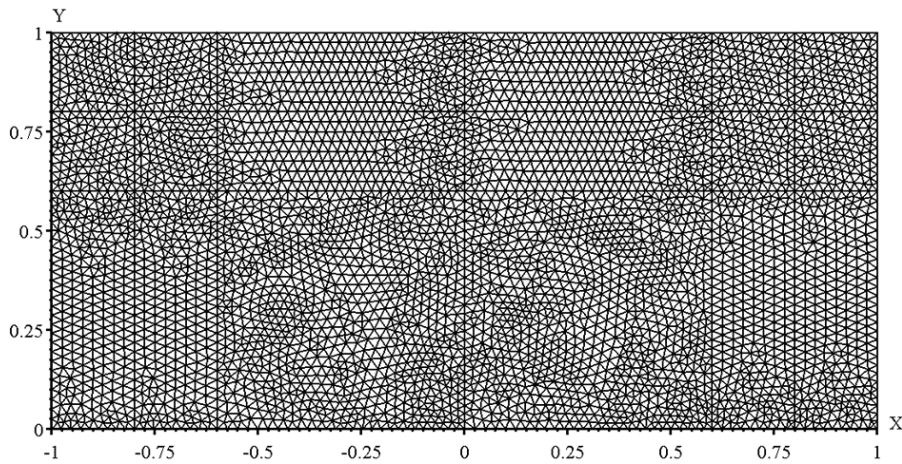


Figure 46. Physical domain and unstructured triangular mesh used for the pure convection of a double-step profile in a rotational velocity field.

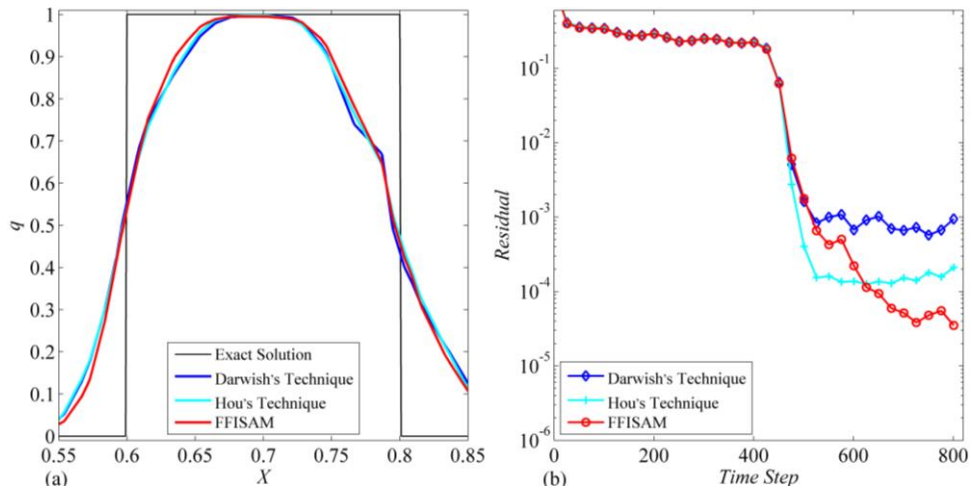


Figure 47. Comparison of accuracy and convergence for various r -factor algorithms ($Cu=1.5$) in Test 4 (Koren / Double-step Rotation / Unstructured triangular grid): (a) scalar profiles; (b) decay of the norm of the residuals.

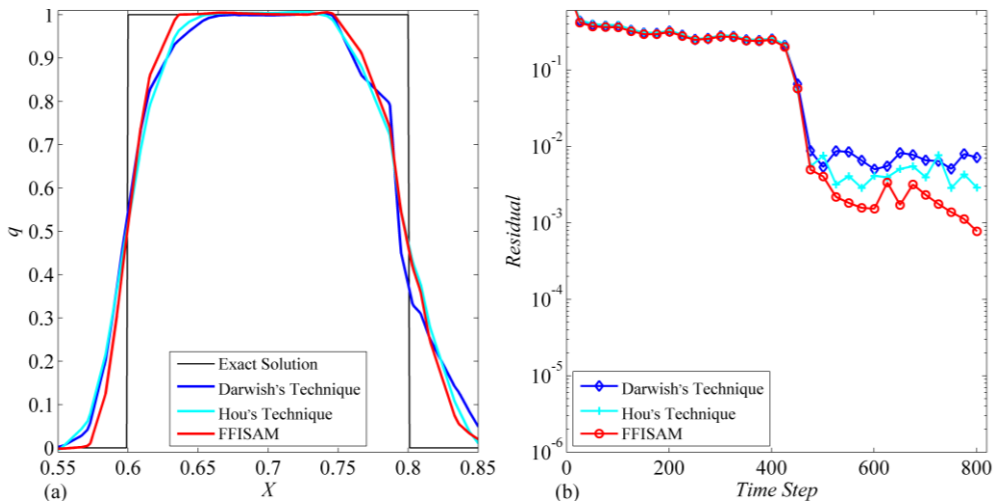


Figure 48. Comparison of accuracy and convergence for various r -factor algorithms ($Cu=1.5$) in Test 4 (Superbee / Double-step Rotation / Unstructured triangular grid): (a) scalar profiles; (b) decay of the norm of the residuals.

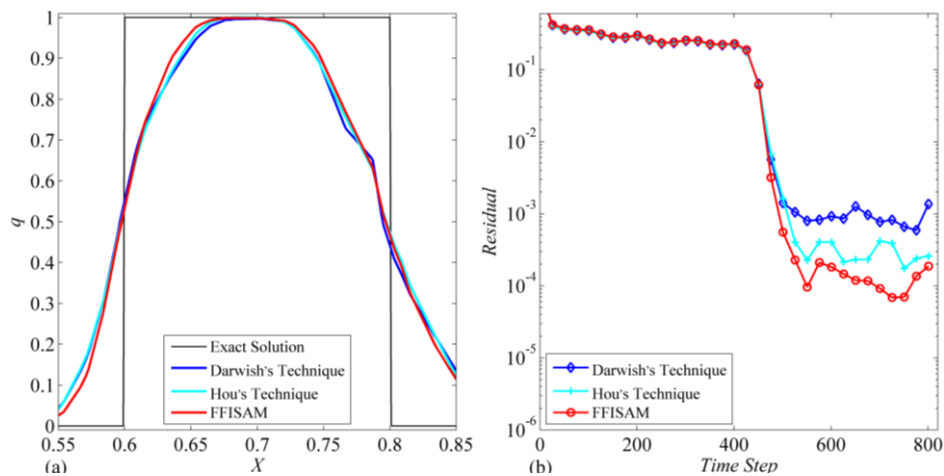


Figure 49. Comparison of accuracy and convergence for various r -factor algorithms ($Cu=1.5$) in Test 4 (MUSCL / Double-step Rotation / Unstructured triangular grid): (a) scalar profiles; (b) decay of the norm of the residuals.

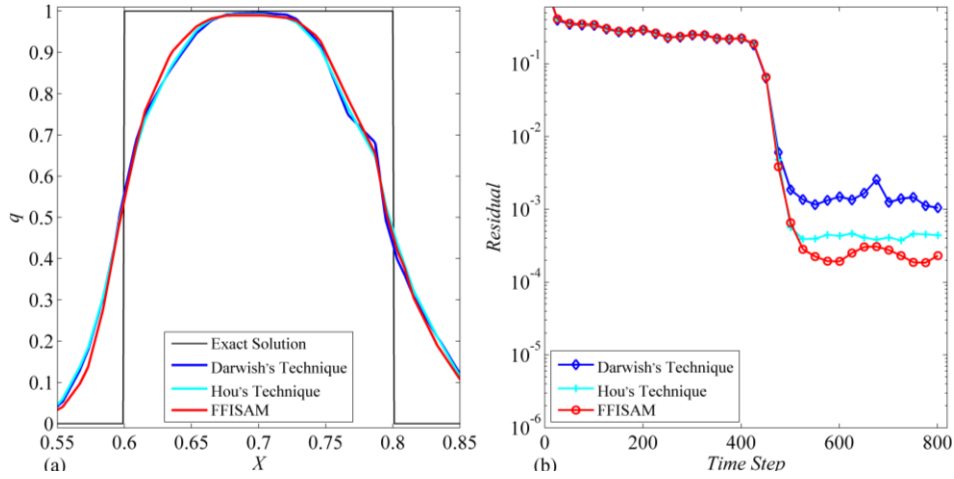


Figure 50. Comparison of accuracy and convergence for various r -factor algorithms ($Cu=1.5$) in Test 4 (WACEB / Double-step Rotation / Unstructured triangular grid): (a) scalar profiles; (b) decay of the norm of the residuals.

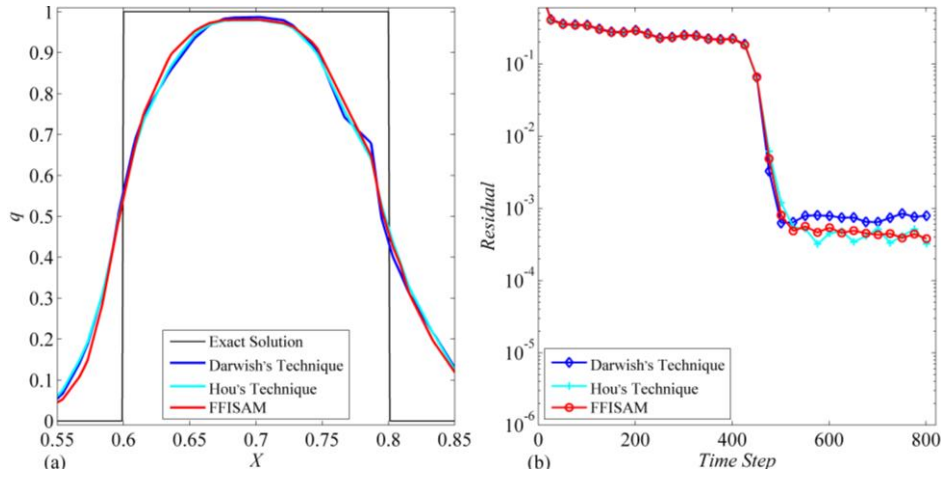


Figure 51. Comparison of accuracy and convergence for various r -factor algorithms ($Cu=1.5$) in Test 4 (TCDF / Double-step Rotation / Unstructured triangular grid): (a) scalar profiles; (b) decay of the norm of the residuals.

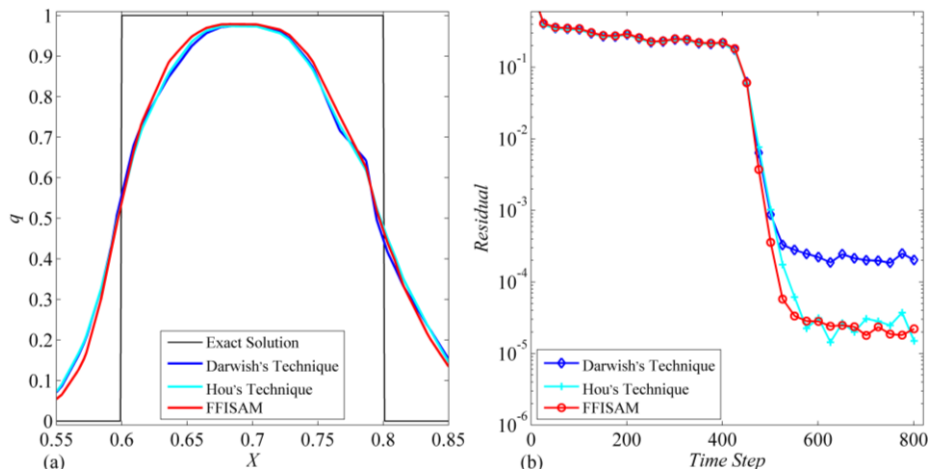


Figure 52. Comparison of accuracy and convergence for various r -factor algorithms ($Cu=1.5$) in Test 4 (Harmonic / Double-step Rotation / Unstructured triangular grid): (a) scalar profiles; (b) decay of the norm of the residuals.