INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS
Int. J. Numer. Meth. Fluids 2003; **41**:997–1019 (DOI: 10.1002/fld.483)

The third-order polynomial method for two-dimensional convection and diffusion

Pavlo Tkalich*,† and Eng Soon Chan

Tropical Marine Science Institute, National University of Singapore, 14 Kent Ridge Road, Singapore 119223, Singapore

SUMMARY

Using the upstream polynomial approximation a series of accurate two-dimensional explicit numerical schemes is developed for the solution of the convection–diffusion equation. A third-order polynomial approximation (TOP) of the convection term and a consistent second-order approximation of the diffusion term are combined in a single-step flux-difference algorithm. Stability analysis confirms that the TOP-12 scheme satisfies the CFL condition for two dimensions. Using smaller and narrower flux stencils compared to algorithms of similar accuracy, the TOP-12 scheme is more efficient in terms of computations per single node. Numerical tests and comparison with other well-known algorithms show a high performance of the developed schemes. Copyright © 2003 John Wiley & Sons, Ltd.

KEY WORDS: third order; convection; diffusion; conservative; pseudo-flux; polynomial interpolation

1. INTRODUCTION

The solution of many problems of fluid dynamics typically involves the numerical approximation of an unsteady convection–diffusion equation

$$\frac{\partial C}{\partial t} = -\nabla(C\mathbf{u}) + \nabla(\mathbf{E}\nabla C) \tag{1}$$

Here $\mathbf{u}=(u^x,u^y)$ is the fluid velocity in x and y direction, respectively; $\mathbf{E}=(E^x,E^y)$ is the turbulent eddy coefficient; $C=C(t,x,y)$ is the transported scalar quantity; $\nabla=(\partial/\partial x,\partial/\partial y)$; and t is the time variable. A control-volume form of Equation (1) for a quasi-uniform flow can be written as

$$C_{i,j}^{k+1} = C_{i,j}^{k} - \alpha(C_{\mathrm{E}} - C_{\mathrm{W}}) - \beta(C_{\mathrm{N}} - C_{\mathrm{S}}) + \lambda(C'_{\mathrm{E}} - C'_{\mathrm{W}}) + \mu(C'_{\mathrm{N}} - C'_{\mathrm{S}}) \tag{2}$$

where $C_{i,j}^{k}$ is the cell-averaged value of the transported scalar at the central (i,j) computational cell at time-level k; C_{W}, C_{E}, C_{S}, C_{N} are the west-, east-, south- and north-face

* Correspondence to: P. Tkalich, Tropical Marine Science Institute, National University of Singapore, 14 Kent Ridge Road, Singapore 119223, Singapore.
† E-mail: tmspt@nus.edu.sg

Received 21 May 2002
Revised 9 December 2002

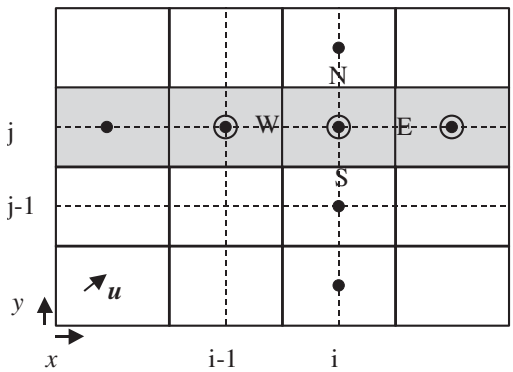

Figure 1. Computational stencil for Davis and Moore [3] scheme (black dots). Nodes for the east-face convective flux (for **u** as shown) are marked with circles. Shaded cells are involved in differencing along the x-axis.

values (convective fluxes) of the transported scalar C for the central cell; C'_{W}, C'_{E}, C'_{S}, C'_{N} are the respective diffusive fluxes; $\alpha = u^x \Delta t/\Delta x$ and $\beta = u^y \Delta t/\Delta y$ are the Courant numbers; $\lambda = E^x \Delta t/(\Delta x)^2$ and $\mu = E^y \Delta t/(\Delta y)^2$ are the diffusion numbers; $\Delta x, \Delta y$ is the grid size in x and y directions, respectively; Δt is the time step. Mass conservation is guaranteed if

$$C_{\mathrm{W}}|_{i,j} = C_{\mathrm{E}}|_{i-1,j}, \quad C_{\mathrm{S}}|_{i,j} = C_{\mathrm{N}}|_{i,j-1}, \quad C'_{\mathrm{W}}|_{i,j} = C'_{\mathrm{E}}|_{i-1,j}, \quad C'_{\mathrm{S}}|_{i,j} = C'_{\mathrm{N}}|_{i,j-1} \tag{3}$$

Various conditions at the boundary Γ of a computational domain can easily be specified utilizing the combined convective–diffusive flux (F) along the normal co-ordinate γ. Thus, the Dirichlet type boundary ($C_\Gamma = v_\Gamma$) leads to $C^{k+1}_{i,j} = v_\Gamma$ and $F_\Gamma = F_{\Gamma-1}$; and the Neumann condition ($\partial C_\Gamma/\partial\gamma = v_\Gamma$) is satisfied when $F_\Gamma = F_{\Gamma-1} + v_\Gamma$.

Advantages of discretization of the convection term using the upstream third-order approximation were discussed intensively by Leonard [1]. Even though, the first descriptions of high-order approximation methods appeared in the literature in the late 1960s, the methods were not popular for a long time due to the computer power constraints. The one-dimensional third-order-accurate QUICKEST algorithm, derived by Leonard [2] using a convective integration method, gave a new breakthrough in computational science toward accurate and efficient numerical schemes. Replacing the time derivative in a Taylor expansion with spatial derivatives, Davis and Moore [3] suggested a different way of deriving the QUICKEST scheme. They neglected cross-derivative terms while extending the scheme for two dimensions (Figure 1). Chen and Falconer [4, 5] formulated various implicit and explicit forms of the two-dimensional QUICKEST schemes. They showed that simple addition of individual x and y contributions of one-dimensional high-order schemes, without appropriate spatial cross-terms, could lead to a basic instability. This was resolved by Leonard and Niknafs [6] and Rasch [7] (respective computational stencil is shown in Figure 2), Ekebjærg and Justesen [8] (Figure 3), and LeVeque's [9] method 5, who included some cross-terms to improve stability of the two-dimensional schemes. The schemes utilize 10-node computational stencils, a formally needed minimum for the third-order accuracy. Although the cross-terms do not affect a formal third-order accuracy, they do improve the stability of the schemes. To achieve

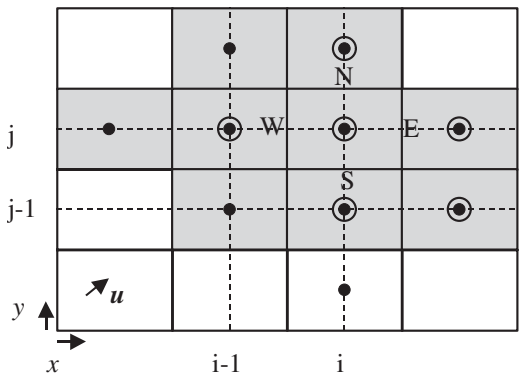

Figure 2. Computational stencil for Leonard and Niknafs [6] and Rasch [7] schemes (black dots). Nodes for the east-face convective flux (for **u** as shown) are marked with circles. Shaded cells are involved in differencing along the x-axis.

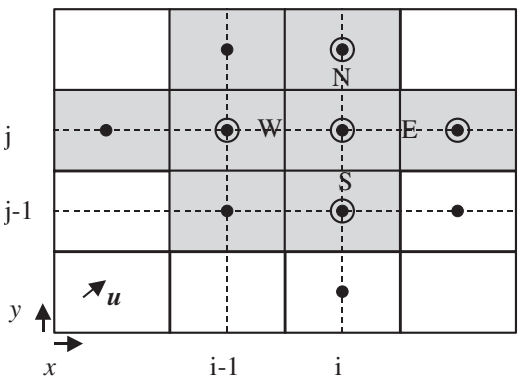

Figure 3. Computational stencil for Ekebjærg and Justesen [8] scheme (black dots). Nodes for the east-face convective flux (for **u** as shown) are marked with circles. Shaded cells are involved in differencing along the x-axis.

better stability and accuracy, Leonard *et al.* [10] employed the cross-terms of up to a fourth-order approximation to develop a 'uniformly third-order polynomial interpolation algorithm' (UTOPIA). The UTOPIA utilizes a 12-node computational stencil and eight-node convective and diffusive fluxes, as shown in Figure 4. LeVeque's [9] method 6 is similar to UTOPIA for the convective term; however, it does not consider diffusive fluxes.

In the effort to refine computational accuracy, one has to cope with a larger number of nodes and computations, associated with a higher-order scheme. The number of nodes in a *computational stencil* cannot be decreased below a certain value without compromising accuracy or stability; however, the reduction of nodes in a *flux stencil* does not necessarily affect properties of the scheme, as long as the entire computational stencil and the resulting single-step update are preserved.

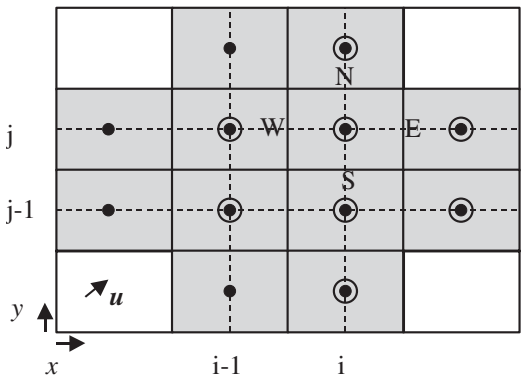

Figure 4. Computational stencil for Leonard *et al*. [10] scheme (black dots). Nodes for the east-face flux (for **u** as shown) are marked with circles. Shaded cells are involved in differencing along the *x*-axis.

It is generally recognized [10] that to maintain the same accuracy as lower-order schemes, higher-order ones allow a much coarser mesh, thus keeping comparable low overall cost. This is not the case for computations in water bodies having narrow channels, where the flux stencils derived in References [6–9] would require at least three computational nodes across a channel, and UTOPIA even four (as in Figures 2–4). Hence, to be efficient a high-order scheme must utilize a narrow flux stencil having a minimum number of nodes. These considerations were taken into account in the paper to develop a set of third-order-accurate numerical schemes using the upstream third-order polynomial method (TOP). The schemes are tested against analytical solutions and other algorithms.

2. PSEUDO-FLUX-DIFFERENCE FORMULATION

2.1. Convective fluxes

The formulation starts with the approximation of the convection term in Equation (1) using the Lagrangian technique, which assumes that the value of the transported scalar C is constant along the trajectory $[(\bar{r}_{i,j}, t^k), (r_{i,j}, t^{k+1})]$ of a fluid particle, i.e. $C^{k+1}_{i,j} = \bar{C}^k_{i,j}$. Here $\bar{r} = (x_i - \alpha\Delta x, y_j - \beta\Delta y)$ is the upstream point. The value $\bar{C}^k_{i,j}$ can be found using the piece-wise interpolation polynomial, P, resulting in the explicit single-step update

$$C^{k+1}_{i,j} = P = (\mathbf{p} \cdot \bar{\mathbf{S}}) \tag{4}$$

where the vector $\bar{\mathbf{S}} = \{\bar{S}_n\}_{n=1,N}$ comprises of values of the transported scalar C in all nodes of a computational stencil at time-level k; N is the number of nodes in the stencil; $\mathbf{p} = \{p_n(\alpha, \beta)\}_{n=1,N}$; and the operation $(\mathbf{X} \cdot \mathbf{Y})$ is the dot product of the vectors $\mathbf{X}, \mathbf{Y}$. The Mth degree interpolation polynomial may take the following form:

$$P = (\mathbf{b} \cdot \mathbf{f}) \tag{5}$$

Here, N components of the vector $\mathbf{b}(\alpha, \beta)$ are the products of two power functions $\alpha^{m'}$ and $\beta^{m''}$ $(m', m'' = 0, M)$; and the coefficient vector $\mathbf{f} = \{f_n\}_{n=1,N}$ has to be identified. The N equa-

tions, $(\mathbf{b}(\alpha_n,\beta_n)\cdot\mathbf{f})=\bar{S}_n$ $(n=1,N)$, can be derived requiring that the polynomial passes through every node from the computational stencil $\bar{\mathbf{S}}$. The resulting system of linear equations is

$$\mathbf{B}\mathbf{f}=\bar{\mathbf{S}} \tag{6}$$

where the matrix $\mathbf{B}=\{\mathbf{b}(\alpha_n,\beta_n)\}_{n=1,N}$ has been identified earlier. Solution of system (6) can be obtained by finding the inverse matrix $\mathbf{B}^{-1}$, and then

$$\mathbf{f}=\mathbf{B}^{-1}\bar{\mathbf{S}} \tag{7}$$

For computational efficiency, the non-conservative single-step update (4) has to be rewritten in a conservative flux-difference form

$$C_{i,j}^{k+1}=C_{i,j}^{k}-\alpha(C_{\mathrm{E}}-C_{\mathrm{W}})-\beta(C_{\mathrm{N}}-C_{\mathrm{S}}) \tag{8}$$

Some procedures, like the flux integral method [10], give a unique formulation of the fluxes; others may produce non-unique fluxes (or pseudo-fluxes). The latter case does not mean that a successful solution cannot be found; moreover, it gives certain flexibility in imposing of additional desirable features on the scheme. Thus, the east-face and north-face convective fluxes are presumed, respectively, in the form

$$C_{\mathrm{E}}=(\mathbf{A}(\alpha,\beta)\cdot\mathbf{S}_i),\quad C_{\mathrm{N}}=(\mathbf{A}(\beta,\alpha)\cdot\mathbf{S}_j) \tag{9}$$

Here the fluxes are specified at stencils $\mathbf{S}_i=\{C_{i',j'}^{k}\}_{l=1,L}$ and $\mathbf{S}_j=\{C_{j',i'}^{k}\}_{l=1,L}$, each comprising of L nodes. Substituting Equations (4) and (9) into the right and left side of Equation (8), respectively, a system of linear algebraic equations can be deduced, where the components $\{A_l\}_{l=1,L}$ have to be identified. The most straightforward splitting of the fluxes is expected for $L=N/2$, when the combined number of unknown values $\{A_l(\alpha,\beta)\}_{l=1,L}$ and $\{A_l(\beta,\alpha)\}_{l=1,L}$ is equal to the number of nodes (N) in the computational stencil. If some equations of the system are linearly dependent, additional conditions have to be employed. Similar procedure is possible for the flux stencils having $L>N/2$ nodes, and the system is inconsistent for $L<N/2$.

The algorithm can be extended for a non-uniform rectangular mesh, though the computational cost (memory mainly) slightly increases. Procedures of computing of coefficients $\mathbf{f}$ (Equation (7)) and the flux splitting (Equations (8)–(9)) have to be repeated for each node once before the main loop. To remain computationally effective, the extended algorithm has to keep the vector $\mathbf{A}$ in memory for every node throughout the entire computational cycle. Modification of the algorithm for a curvilinear mesh would involve the co-ordinate transformation into rectangular ones, and appearance of new terms to be approximated. Influence of the terms on the scheme efficiency is yet to be explored. Three-dimensional version of the scheme can be obtained in a similar fashion.

One may reason that usage of pseudo-fluxes, C_{W}, C_{E}, C_{S} and C_{N}, instead of true fluxes, $C_{\mathrm{W}}^{\mathrm{T}}$, $C_{\mathrm{E}}^{\mathrm{T}}$, $C_{\mathrm{S}}^{\mathrm{T}}$ and $C_{\mathrm{N}}^{\mathrm{T}}$, introduces a lower-order error in a single-step update. This may occur if the scheme formulation is started with the substitution of centroidal values of fluxes instead of cell-averaged ones into Equation (2). However, the suggested in this paper flux-splitting method ensures that the lower-order errors (if any) are mutually cancelled. Indeed, let us assume (for a one-dimensional case) that the eastward pseudo-flux is derived with an error δ, i.e. $C_{\mathrm{E}}=C_{\mathrm{E}}^{\mathrm{T}}+\delta$. Since, the flux-splitting procedure (8)–(9) of the single-step update (4) is formulated to satisfy the equality $C_{\mathrm{E}}-C_{\mathrm{W}}=C_{\mathrm{E}}^{\mathrm{T}}-C_{\mathrm{W}}^{\mathrm{T}}$, a deviation of the westward pseudo-flux

from the 'true' value can be found as $C_W - C_W^T = \delta$. Thus, the introduced lower-order error (if any) is cancelled during the flux differencing, and accuracy of the original single-step update is preserved.

2.2. Diffusive fluxes

The flux-difference formulation of diffusion term from Equation (1) reads

$$C_{i,j}^{k+1} = C_{i,j}^{k} + \lambda(C'_E - C'_W) + \mu(C'_N - C'_S) \tag{10}$$

where east- and north-face diffusive fluxes are sought in the form

$$C'_E = (\mathbf{D}(\alpha,\beta)\cdot\mathbf{S}_i) \quad \text{and} \quad C'_N = (\mathbf{D}(\beta,\alpha)\cdot\mathbf{S}_j) \tag{11}$$

respectively. Here components of the vector $\mathbf{D} = \{D_l\}_{l=1,L}$ are the functions of the Courant numbers α and β. Using Equations (4)–(5), the differences of the diffusive fluxes can be expressed as follows:

$$\begin{aligned} C'_E - C'_W &= (\Delta x)^2 \left.\frac{\partial^2 C}{\partial x^2}\right|_{r=\bar{r}_{i,j}} = \frac{\partial^2 P}{\partial \alpha^2} \\ C'_N - C'_S &= (\Delta y)^2 \left.\frac{\partial^2 C}{\partial y^2}\right|_{r=\bar{r}_{i,j}} = \frac{\partial^2 P}{\partial \beta^2} \end{aligned} \tag{12}$$

At this stage the pseudo-flux-splitting procedure, introduced in Section 2.1, is used. Finally, one can substitute Equations (8)–(11) into (2) to obtain the explicit finite-difference approximation of the convection–diffusion equation (1) as

$$C_{i,j}^{k+1} = C_{i,j}^{k} + F_E - F_W + F_N - F_S \tag{13}$$

where

$$\begin{aligned} F_E &= ((\lambda\mathbf{D}(\alpha,\beta) - \alpha\mathbf{A}(\alpha,\beta))\cdot\mathbf{S}_i) \\ F_N &= ((\mu\mathbf{D}(\beta,\alpha) - \beta\mathbf{A}(\beta,\alpha))\cdot\mathbf{S}_j) \end{aligned} \tag{14}$$

$$F_W|_{i,j} = F_E|_{i-1,j}, \quad F_S|_{i,j} = F_N|_{i,j-1} \tag{15}$$

2.3. Stability analysis

It is instructive to obtain stability criteria of newly developed numerical schemes. According to the von Neumann stability analysis, a solution of Equation (13) is sought in the form

$$C_{i,j}^{k} = V^k \exp(I(i\theta^x + j\theta^y))$$

where $\theta^x = k^x\Delta x$, $\theta^y = k^y\Delta y$ are the phase angles; V^k is the amplitude of the Fourier component at $k\Delta t$; k^x, k^y are the wave numbers; $I = \sqrt{-1}$. The method requires that the amplification factor over one complete time step satisfy the following condition for all possible values of θ^x, θ^y:

$$|G(\theta)| = |V^{k+1}/V^k| \leqslant 1 \tag{16}$$

For all schemes developed in this paper, the amplification factor is computed and analysed graphically for the special case of uniform flow and grid ($\alpha, \beta = \text{const}$, $\sigma = \lambda = \mu$ and $\theta = \theta^x = \theta^y$).

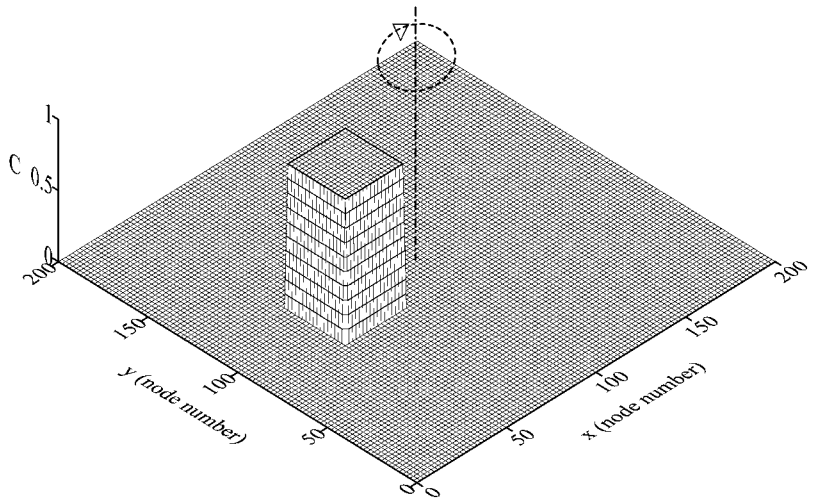

Figure 5. Initial shape (rectangular parallelepiped) of the transported scalar for the test case.

2.4. Test case

To verify the TOP schemes, a series of numerical tests is performed. One of the tests is a clockwise rotation of a transported scalar field, shaped as a rectangular parallelepiped (Figure 5). It is expected that for a pure convection an ideal numerical scheme does not introduce a distortion into the initial shape, while moving it with the flow. The idealized 2D constant-depth domain consists of 201×201 grid cells with the grid size $(\Delta x \times \Delta y) = (1 \times 1)$ km. Zero fluxes are assumed at the four boundaries. The initial non-zero transported scalar values, $C^0 = 1$, are located within the rectangle with bottom-left and top-right corners $(44\Delta x, 84\Delta y)$ and $(78\Delta x, 118\Delta y)$, respectively. The time step was chosen to give a maximal Courant number 0.8 for UTOPIA and TOP-12 schemes, and 0.4 for TOP-10 (A and B). For the scheme comparison, an error measure at the rectangular grid with the grid size Δx is introduced as

$$\varepsilon(\Delta x) = \frac{\sum_{i,j} |C_{i,j} - C^*_{i,j}|}{\sum_{i,j} |C^*_{i,j}|} \tag{17}$$

where C^* is the exact solution, and C is the numerical one. Computations are conducted to allow five complete rotations of the profile, and then the final extrema $(C_{\min}, C_{\max})$, and the error measures are compared in Table I.

3. THIRD-ORDER POLYNOMIAL METHOD

The higher the degree of the polynomial (5), the higher the accuracy of the resulting computational scheme. The maximum accuracy and stability within the class of third-degree

Table I. Comparison of main features and performance of UTOPIA and TOP schemes for the pure convection test.

Scheme	Maximum Courant numbers	Number of nodes: across flux stencil/in flux stencil/in computational stencil	C_{max}	C_{min}	Error, ε
UTOPIA	$\lvert\alpha\rvert=1,\ \lvert\beta\rvert=1$	4/8/12	1.109	−0.057	0.32546
TOP-12	$\lvert\alpha\rvert=1,\ \lvert\beta\rvert=1$	2/6/12	1.110	−0.060	0.32567
TOP-10A	$\lvert\alpha\rvert+\lvert\beta\rvert=1$	2/5/10	1.113	−0.061	0.33692
TOP-10B	$\lvert\alpha\rvert+\lvert\beta\rvert=1$	2/5/10	1.111	−0.061	0.32664

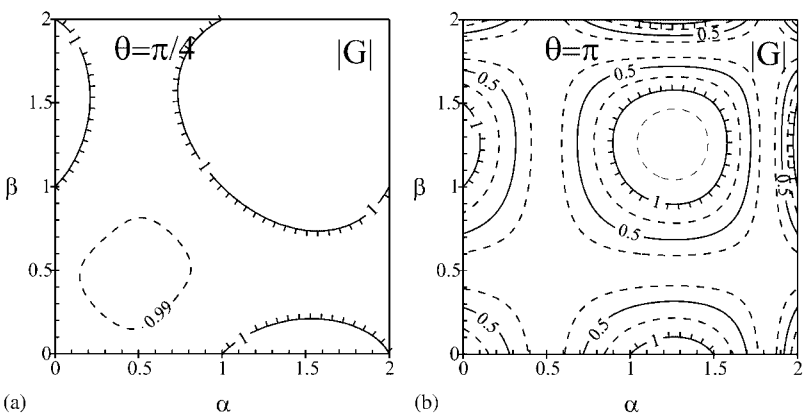

Figure 6. A modulus of amplification factor for TOP-16 scheme.

polynomials ($M=3$) can be achieved by retaining all 16 terms, leading to

$$P(\alpha,\beta)=\begin{pmatrix} \overset{0}{f_{00}} & \overset{\mathrm{I}}{+f_{10}\alpha} & \overset{\mathrm{II}}{+f_{20}\alpha^2} & \overset{\mathrm{III}}{+f_{30}\alpha^3+} \\ +f_{01}\beta & +f_{11}\alpha\beta & +f_{21}\alpha^2\beta & -+f_{31}\alpha^3\beta+ \\ +f_{02}\beta^2 & +f_{12}\alpha\beta^2 & -+f_{22}\alpha^2\beta^2 & +f_{32}\alpha^3\beta^2+ \\ +f_{03}\beta^3 & -+f_{13}\alpha\beta^3 & +f_{23}\alpha^2\beta^3 & +f_{33}\alpha^3\beta^3 \\ \underset{\mathrm{IV}}{} & & \underset{\mathrm{V}}{} & \underset{\mathrm{VI}}{} \end{pmatrix} \tag{18}$$

In Equation (18) shaded terms contribute into individual x- or y-direction, while rest terms represent mixed x- and y-contribution (equivalent to the spatial cross-derivative terms in the Taylor-series expansion). Diagonal bends identify terms of a similar order, shown by Roman numerals. A single-step update for scheme (4), (18) is given in Appendix A.1, and the computed amplification factor for the selected values of phase angle θ is presented in Figure 6. The von Neumann stability analysis shows that the method is stable for $|\alpha|\leqslant 1, |\beta|\leqslant 1$. Curiously, the method is stable far beyond the CFL condition, if one of the Courant numbers assumes a value in the vicinity of 0.5. However, the method is inefficient due to a large number of nodes used in the computational stencil. Following the procedure presented in

Section 2, several more efficient third-order schemes can be developed. To obtain a barely third-order accurate scheme, one has to retain all terms of up to third order (ten terms above the dash–dot line in the polynomial (18)). The scheme would require a computational stencil, having at least 10 nodes, as shown in Figures 2 and 3. It is expected that schemes utilizing less cross-terms in polynomial (18) (and less number of nodes, respectively) can only be second-order accurate, as the eight-node MOSQUITO scheme [11]. Generally, configuration of the computational stencil and a number of nodes in it are not unique, because different methods of the scheme derivation may produce different results. For example, the 11-node STOUS scheme [12] utilizes the same-order cross-terms (and almost as accurate) as the 10-node schemes of Leonard and Niknafs [6], Rasch [7], Ekebjærg and Justesen [8]. A number of nodes and configuration of flux stencils are not unique as well (compare the five-node flux stencil of Ekebjærg and Justesen [8], and six-node one of Leonard and Niknafs [6], Rasch [7], Gerges and McCorquodale [12]). Even a width of the stencils is not unique, in spite the fact that all the above-mentioned flux stencils are three-node wide. Addition of fourth- or higher-order cross-terms in polynomial (18) does not improve the formal accuracy of the third-order scheme; however, they do improve the phase behaviour and stability. Thus, the third-order UTOPIA [10] algorithm, considering additionally two cross-terms of fourth order, $0(\alpha^3\beta)$ and $0(\alpha\beta^3)$, widens the typical for the 10-node schemes stability criterion from $|\alpha|+|\beta|\leqslant 1$ to $|\alpha|\leqslant 1, |\beta|\leqslant 1$. The price paid for the extended stability was increased number of nodes within the computational stencil (twelve nodes), in the flux stencil (eight), and increased width of the stencil (four nodes). Even though, utilized in the UTOPIA derivation flux integral method produces a unique solution, some other non-unique methods may yield more compact and efficient schemes. In the following sections, the procedure presented in Section 2 is used to derive a series of two-dimensional third-order schemes having 12-node and 10-node computational stencils. For a convenience of reference, the methods are identified as TOP-12 and TOP-10, respectively. The schemes are concisely formulated in terms of $\mathbf{A}, \mathbf{D}, \mathbf{S}_i, \mathbf{S}_j$, which one can easily substitute into Equations (13)–(14) to obtain the explicit expressions for the convective and diffusive fluxes. Additionally, intermediate and alternative formulations are given in Appendix A.

3.1. TOP-12 method

Retaining terms of up to fourth-order in polynomial (18) (12 terms above a solid line), and utilizing a 12-node computational stencil and a six-node flux stencil as in Figure 7, one obtains

$$\begin{aligned}
\mathbf{S}_i &= (C^k_{i+1,j}, C^k_{i,j}, C^k_{i-1,j}, C^k_{i+1,j-1}, C^k_{i,j-1}, C^k_{i-1,j-1})\\
\mathbf{S}_j &= (C^k_{i,j+1}, C^k_{i,j}, C^k_{i,j-1}, C^k_{i-1,j+1}, C^k_{i-1,j}, C^k_{i-1,j-1})\\
\mathbf{A}(\phi,\psi) &= ((1-\phi)(2-\phi)(1-\psi)/6, ((1+\phi)(5-2\phi)\\
&\quad -\psi(2+3\phi-2\phi^2))/6, -(1-\phi^2)(1-\psi)/6,\\
&\quad \psi(1-\phi)(2-\phi)/6, \psi(2+3\phi-2\phi^2)/6, -\psi(1-\phi^2)/6)\\
\mathbf{D}(\phi,\psi) &= ((1-\phi)(1-\psi), -(1-2\phi)(1-\psi),\\
&\quad -\phi(1-\psi), (1-\phi)\psi, -(1-2\phi)\psi, -\phi\psi)
\end{aligned} \tag{19}$$

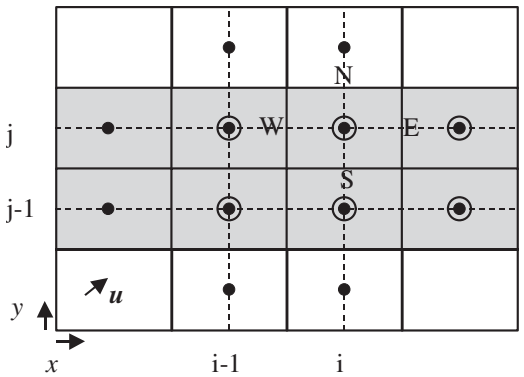

Figure 7. Computational stencil for the TOP-12 scheme (black dots). Nodes for the east-face convective flux (for **u** as shown) are marked with circles. Shaded cells are involved in differencing along the x-axis.

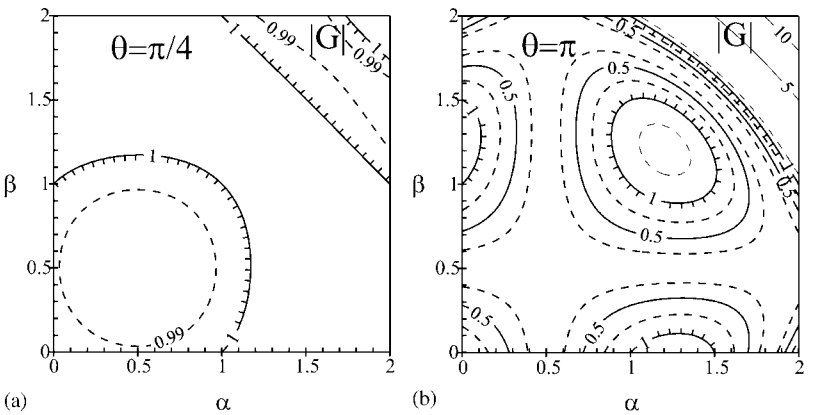

Figure 8. A modulus of amplification factor for TOP-12 scheme.

The respective east-face convective flux for the TOP-12 scheme is

$$\begin{aligned} C_{\mathrm{E}} = {} & \tfrac{1}{6}(1-\alpha)(2-\alpha)(1-\beta)C^k_{i+1,j} - \tfrac{1}{6}(1-\alpha^2)(1-\beta)C^k_{i-1,j} \\ & + \tfrac{1}{6}((5+3\alpha-2\alpha^2)-\beta(2+3\alpha-2\alpha^2))C^k_{i,j} + \tfrac{1}{6}(1-\alpha)(2-\alpha)\beta C^k_{i+1,j-1} \\ & + \tfrac{1}{6}(2+3\alpha-2\alpha^2)\beta C^k_{i,j-1} - \tfrac{1}{6}(1-\alpha^2)\beta C^k_{i-1,j-1} \end{aligned} \tag{20}$$

Some intermediate steps of the TOP-12 derivation, as well as a single-step update for the scheme are given in Appendix A.2. The von Neumann condition defines a stability region for the scheme as $|\alpha| \leqslant 1,\ |\beta| \leqslant 1,\ 0 \leqslant \lambda \leqslant 0.25,\ 0 \leqslant \mu \leqslant 0.25$, which is similar to that of the UTOPIA algorithm [10]. The amplification factor for the convective scheme is plotted in Figures 8 and 9 for different values of the Courant numbers and phase angle. Using the TOP-12 scheme for the convective term approximation, the result of the rotation test computation is shown in Figure 10, and the respective extrema values and the error measures are given in Table I. The UTOPIA algorithm demonstrates almost identical results for the test. In fact,

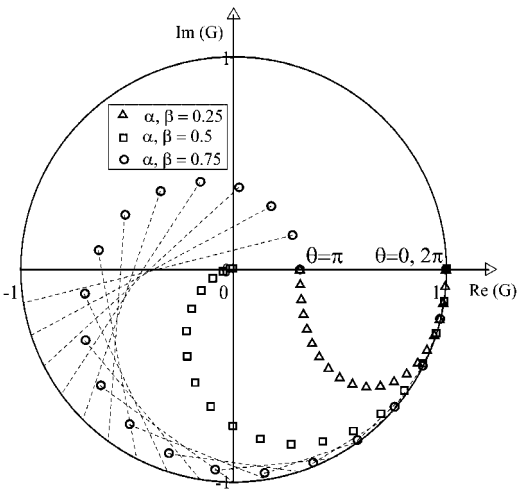

Figure 9. Polar plot of amplification factor for TOP-12 scheme for the different values of the Courant numbers.

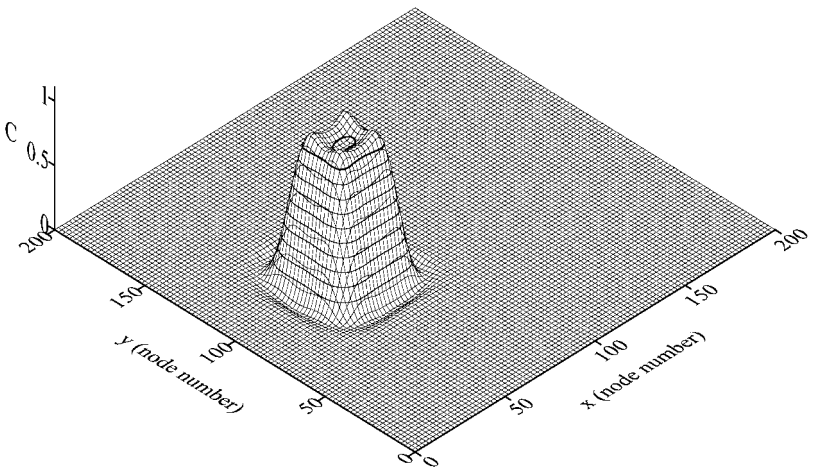

Figure 10. Test case for a pure convection of rectangular parallelepiped. Computed scalar field after five complete rotations using TOP-12 scheme.

the final profiles are so close to each other that even the wiggle patterns are indistinguishable (Figure 11). Using 25% less nodes for the flux expression than UTOPIA, the TOP-12 scheme is more efficient per a single node computation.

Diffusive flux of the TOP-12 scheme utilizes the same (six node) stencil as the convective one. The convective and diffusive fluxes are easily combined to form an efficient single-time-step algorithm. An example of the algorithm application for the test case (with the uniform diffusion coefficient $E = 5\text{m}^2/\text{s}$) is given in Figure 12. Numerical computations for different Courant and diffusion numbers show robustness of the combined scheme. One may notice that diffusive flux of the TOP-12 scheme can be presented as the bilinear upwind

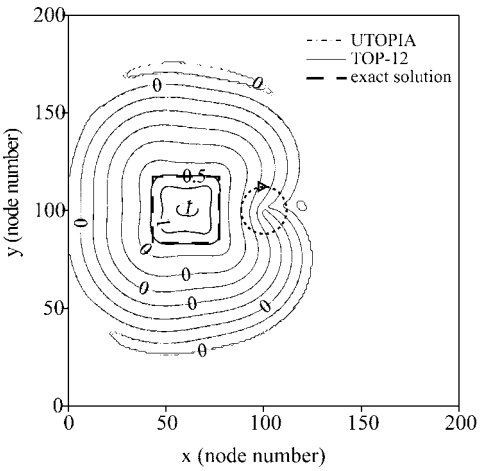

Figure 11. Test case for a pure convection of rectangular parallelepiped. Comparison of the computed scalar field after five complete rotations using TOP-12 and UTOPIA schemes.

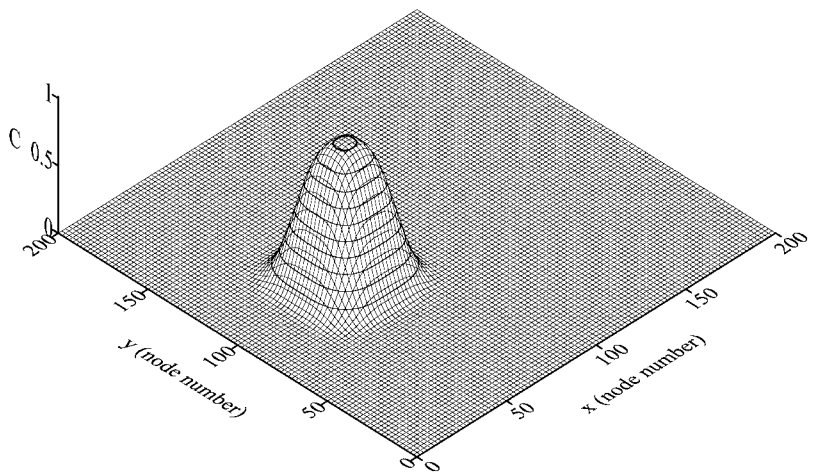

Figure 12. Test case for a convection and diffusion of rectangular parallelepiped. Computed scalar field after five complete rotations using TOP-12 scheme.

interpolation between the fluxes of the forward-time-central-space-difference (FTCS) scheme ($C'_{\mathrm{E}}|^{\mathrm{FTCS}}_{i,j} = C^k_{i+1,j} - C^k_{i,j}$), i.e. for positive **u**

$$C'_{\mathrm{E}}|^{\mathrm{TOP\text{-}12}}_{i,j} = (1-\alpha)(1-\beta)C'_{\mathrm{E}}|^{\mathrm{FTCS}}_{i,j} + \alpha(1-\beta)C'_{\mathrm{E}}|^{\mathrm{FTCS}}_{i-1,j} + (1-\alpha)\beta C'_{\mathrm{E}}|^{\mathrm{FTCS}}_{i,j-1} + \alpha\beta C'_{\mathrm{E}}|^{\mathrm{FTCS}}_{i-1,j-1}$$

3.2. *TOP-10 schemes*

Neglecting all fourth-order cross-terms, one may consider more simple cases of polynomial (18) (remaining ten terms are underlined by a dash–dot line). To define the polynomial coefficients for such a case, a 10-node computational stencil is required. This is the minimum

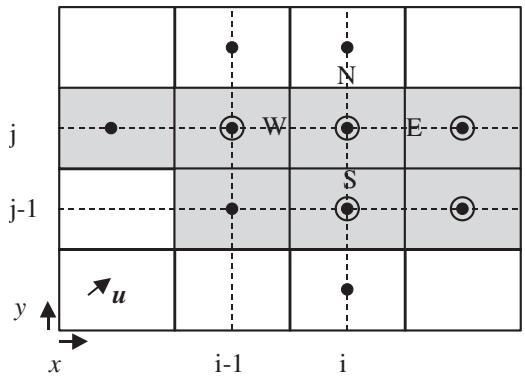

Figure 13. Computational stencil for the TOP-10A scheme (black dots). Nodes for the east-face convective flux (for **u** as shown) are marked with circles. Shaded cells are involved in differencing along the x-axis.

needed for the formal third-order accuracy in two dimensions. Using the algorithm elaborated in Section 2, two schemes having five-node flux stencils are derived.

Scheme A: For the stencil shown in Figure 13, the scheme can be derived as

$$\begin{aligned}
\mathbf{S}_i &= (C^k_{i+1,j}, C^k_{i,j}, C^k_{i-1,j}, C^k_{i+1,j-1}, C^k_{i,j-1}) \\
\mathbf{S}_j &= (C^k_{i,j+1}, C^k_{i,j}, C^k_{i,j-1}, C^k_{i-1,j+1}, C^k_{i-1,j}) \\
\mathbf{A}(\phi,\psi) &= ((1-\phi)(2-\phi-3\psi)/6, (1+\phi)(5-2\phi)/6 - \phi\psi/2 \\
&\quad -(1-\phi^2)/6, (1-\phi)\psi/2, \phi\psi/2) \\
\mathbf{D}(\phi,\psi) &= (1-\phi-\psi, 2\phi-1+\psi, -\phi, \psi, -\psi)
\end{aligned} \tag{21}$$

The respective east-face convective flux is

$$\begin{aligned}
C_{\mathrm{E}} = {} & \tfrac{1}{6}\alpha(1-\alpha)(2-\alpha-3\beta)C^k_{i+1,j} + \tfrac{1}{6}\alpha((5+3\alpha-2\alpha^2)-3\alpha\beta)C^k_{i,j} \\
& -\tfrac{1}{6}\alpha(1-\alpha^2)C^k_{i-1,j} + \tfrac{1}{2}\alpha\beta(1-\alpha)C^k_{i+1,j-1} + \tfrac{1}{2}\alpha^2\beta C^k_{i,j-1}
\end{aligned} \tag{22}$$

Scheme B: For the stencil shown in Figure 14, the scheme is developed as

$$\begin{aligned}
\mathbf{S}_i &= (C^k_{i+1,j}, C^k_{i,j}, C^k_{i-1,j}, C^k_{i,j-1}, C^k_{i-1,j-1}) \\
\mathbf{S}_j &= (C^k_{i,j+1}, C^k_{i,j}, C^k_{i,j-1}, C^k_{i-1,j}, C^k_{i-1,j-1}) \\
\mathbf{A}(\phi,\psi) &= ((1-\phi)(2-\phi)/6, (1+\phi)(5-2\phi)/6 - \psi(2-\phi)/2, \\
&\quad -(1-\phi)(1+\phi-3\psi)/6, (2-\phi)\psi/2, -(1-\phi)\psi/2) \\
\mathbf{D}(\phi,\psi) &= (1-\phi, 2\phi-1-\psi, \psi-\phi, \psi, -\psi)
\end{aligned} \tag{23}$$

The respective east-face convective flux for the scheme is given by

$$\begin{aligned}
C_{\mathrm{E}} = {} & \tfrac{1}{6}\alpha(1-\alpha)(2-\alpha)C^k_{i+1,j} + \tfrac{1}{6}\alpha((1+\alpha)(5-2\alpha)-3\beta(2-\alpha))C^k_{i,j} \\
& -\tfrac{1}{6}\alpha(1-\alpha)(1+\alpha-3\beta)C^k_{i-1,j} + \tfrac{1}{2}\alpha\beta(2-\alpha)C^k_{i,j-1} - \tfrac{1}{2}\alpha\beta(1-\alpha)C^k_{i-1,j-1}
\end{aligned} \tag{24}$$

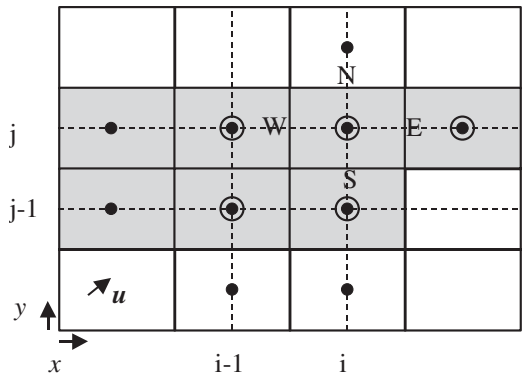

Figure 14. Computational stencil for the TOP-10B scheme (black dots). Nodes for east-face convective flux (for **u** as shown) are marked with circles. Shaded cells are involved in differencing along the x-axis.

Some intermediate expressions for the TOP-10 schemes derivation are given in Appendices A.3 and A.4. Plots of the amplification factor for the TOP-10A scheme are given in Figures 15 and 16. The von Neumann analysis identifies the stability region for the schemes as $|\alpha| + |\beta| \leqslant 1,\ 0 \leqslant \lambda \leqslant 0.25,\ 0 \leqslant \mu \leqslant 0.25$. Having the single-step updates similar to some of the algorithms from References [6–8], the TOP-10 schemes are using narrower flux stencils with fewer nodes (compare Figures 13 and 14 with Figures 2 and 3). Even though, the computational cost of TOP-10 schemes per one node is less than that of TOP-12, overall the latter scheme is more efficient because it allows for larger time step. The rotation test for the TOP-10 (A and B) schemes show slightly more wiggling behaviour, but still have a comparable accuracy with TOP-12 and UTOPIA (see Table I); though the differences are hardly noticeable (Figure 17). This confirms the finding of Leonard *et al.* [10] that addition of cross-terms into a scheme does not improve accuracy in as much as stability. All the compared schemes are strictly mass conservative, i.e.

$$\sum_{i,j} C_{i,j} \Big/ \sum_{i,j} C^*_{i,j} = 1$$

With the cross-terms omitted altogether in polynomial (18) (only shaded seven terms remain), the Davis and Moore [3] scheme follows:

$$\begin{aligned} \mathbf{S}_i &= (C^k_{i+1,j}, C^k_{i,j}, C^k_{i-1,j}), \quad \mathbf{S}_j = (C^k_{i,j+1}, C^k_{i,j}, C^k_{i,j-1}) \\ \mathbf{A}(\phi,\psi) &= ((1-\phi)(2-\phi)/6, (1+\phi)(5-2\phi)/6, -(1-\phi^2)/6) \\ \mathbf{D}(\phi,\psi) &= (1-\phi, 2\phi-1, -\phi) \end{aligned} \tag{25}$$

The scheme is unstable for pure convection, except along the axes. In the one-dimensional case the scheme reduces to the QUICKEST algorithm.

Due to the flux-difference formulation, the developed higher-order explicit schemes guarantee mass conservation in a computational domain and near boundaries. Before computing a

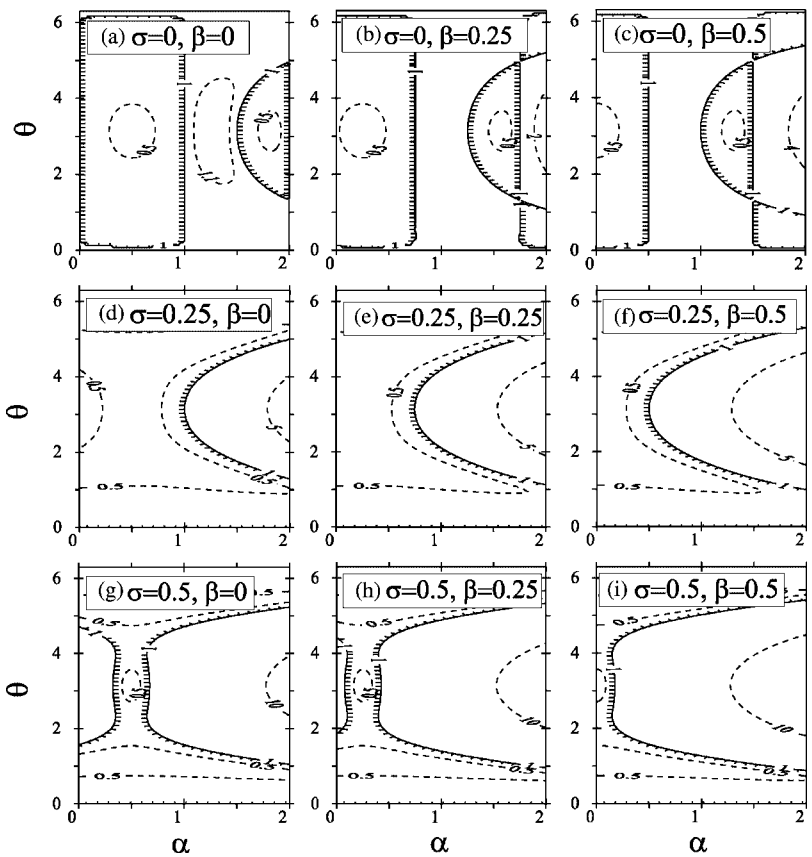

Figure 15. A modulus of amplification factor for TOP-10 schemes, as a function of Courant number α and phase angle θ, for the different values of Courant number β and diffusion number σ.

new node value $C_{i,j}^{k+1}$, one must ensure that every node from the four flux stencils ($\mathbf{S}_i, \mathbf{S}_{i-1}, \mathbf{S}_j$ and $\mathbf{S}_{j-1}$) are within the computational domain; otherwise, the associated with the protruded stencil flux is computed to satisfy boundary conditions. Usage of more compact schemes, such as TOP, may increase number of useful internal computational points by means of delaying the switch to the boundary mode. This is especially obvious for domains having narrow channels along the x- or y-direction. For a comparison, UTOPIA would require a minimum of four nodes per a channel cross-section, other reviewed in this paper schemes need at least three nodes, and all TOP schemes would require only two nodes.

3.3. Accuracy evaluation of the TOP-12 scheme

It was noted [1, 13] that for a one-dimensional case, if Mth-degree piece-wise polynomial is used for the sub-grid interpolation, the resulting convection scheme will be $(M+1)$th-order accurate in both space and time. This was shown analytically for first-, second-, and third-order (QUICKEST) methods. Similar analysis can be done for the TOP-12 scheme, though

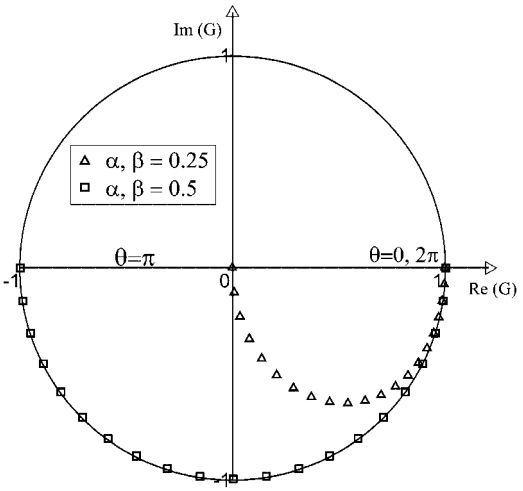

Figure 16. Polar plot of amplification factor for TOP-10 schemes for the different values of the Courant numbers.

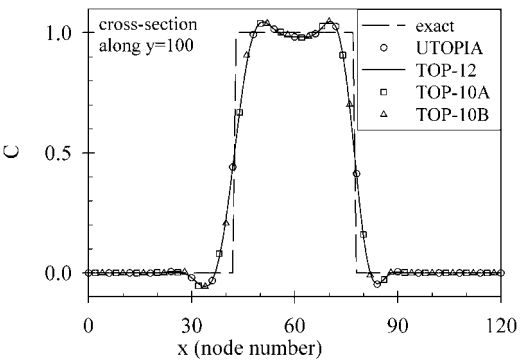

Figure 17. Test case for a pure convection of rectangular parallelepiped. Comparison of cross-sections of the computed scalar field after five complete rotations using UTOPIA, TOP-12, TOP-10A and TOP-10B schemes.

the derivation is complicated by presence of the cross-terms. Leonard [14] suggested (an easier alternative) to use a grid-refinement study in order to show the true convergence rate. According to the method, if the error measure at the rectangular grid with the grid size Δx is introduced as Equation (17), then the convergence rate of the numerical solution to the exact one can be obtained from

$$R = \ln\left(\frac{\varepsilon(\Delta x)}{\varepsilon(\Delta x/2)}\right)\frac{1}{\ln 2} \tag{26}$$

where the rate, R, should approach an asymptotic value as the grid size decreases.

To perform the grid-refinement study, the test case from Section 2.4 is modified as follows. The same size 2D constant-depth domain (200×200) km is covered by a mesh with the grid

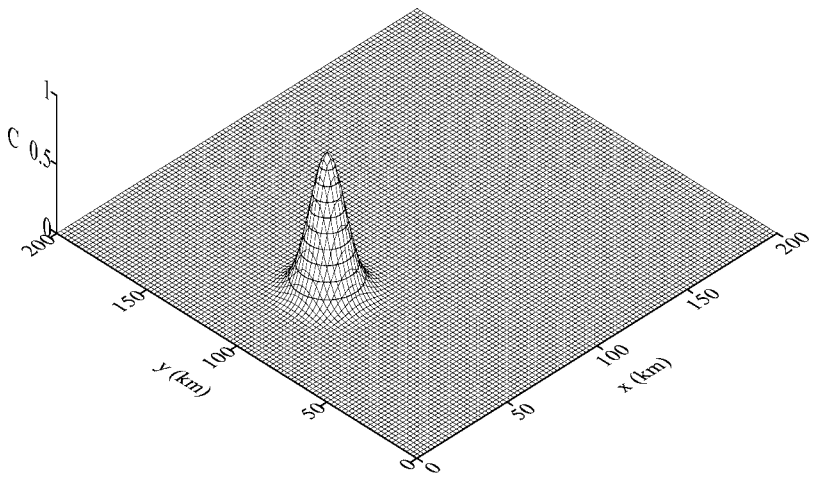

Figure 18. Test case for a pure convection of the Gaussian hill. Initial state.

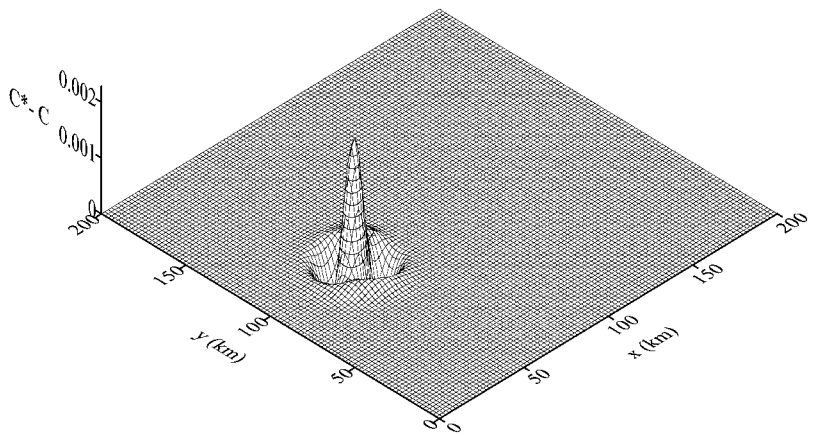

Figure 19. Test case for a pure convection of the Gaussian hill. The difference between exact and computed shapes after five complete rotations.

size $\Delta x = \Delta y = (2, 1, 0.5 \text{ and } 0.25)$ km, leading to the computational grids with (101×101), (201×201), (401×401) and (801×801) nodes, respectively. Clockwise rotation of the Gaussian hill, as in Figure 18, is considered; and the maximal Courant number of 0.8 is maintained through the numerical experiment. The maximal difference between computed and exact values for the grid size of 0.25 km is found to be within of ± 0.002, as in Figure 19. The computed error measures and convergence rates are summarized in Table II. The analysis shows that the TOP-12 scheme is third-order accurate under variable velocity conditions.

Table II. Grid-refinement study of convection terms using the TOP-12 scheme.

Grid size, Δx (km)	Error measure, ε	Convergence rate, R
2	4.189×10^{-1}	—
1	8.955×10^{-2}	2.21
0.5	1.473×10^{-2}	2.60
0.25	1.792×10^{-3}	3.04

4. CONCLUSION

Using the polynomial interpolation method, a series of two-dimensional explicit third-order accurate numerical schemes (TOP) is obtained. For a chosen computational stencil a unique single-step update can be easily derived using the formalized matrix technique. The flux splitting procedure of the update, involving the solution of a set of linear equations, is not necessarily unique. However, the linearly dependent equations can be substituted with other external relationships to introduce desirable properties into the scheme. This feature is utilized in the paper to increase efficiency of high-order schemes by employing narrow flux stencils with reduced number of nodes. Thus, TOP-12 scheme, having a six-node flux stencil, requires less computation per node, than the equally accurate eight-node UTOPIA algorithm. Compared to the four-node-wide flux stencil of UTOPIA, the TOP-12's flux stencil is just two-node-wide, allowing for a coarser mesh to be used in the vicinity of boundaries or in narrow channels. In terms of accuracy and stability the two schemes are similar. The grid resolution test shows that the TOP-12 scheme is third-order accurate. The TOP-10 (A and B) schemes show the same preferable features over equivalent algorithms of other authors.

APPENDIX A

A.1. TOP-16

A single-step update for the TOP-16 convective scheme results in

$$
\begin{aligned}
C_{i,j}^{k+1} = {} & \tfrac{1}{36}\alpha\beta(1-\alpha^2)(1-\beta^2)C_{i-2,j-2}^{k} - \tfrac{1}{12}\alpha\beta(1+\alpha)(2-\alpha)(1-\beta^2)C_{i-1,j-2}^{k} \\
& - \tfrac{1}{12}\beta(2-\alpha)(1-\alpha^2)(1-\beta^2)C_{i,j-2}^{k} + \tfrac{1}{36}\alpha\beta(1-\alpha)(2-\alpha)(1-\beta^2)C_{i+1,j-2}^{k} \\
& - \tfrac{1}{12}\alpha\beta(1-\alpha^2)(1+\beta)(2-\beta)C_{i-2,j-1}^{k} + \tfrac{1}{4}\alpha\beta(1+\alpha)(2-\alpha)(1+\beta)(2-\beta)C_{i-1,j-1}^{k} \\
& + \tfrac{1}{4}\beta(2-\alpha)(1-\alpha^2)(1+\beta)(2-\beta)C_{i,j-1}^{k} - \tfrac{1}{12}\alpha\beta(1-\alpha)(2-\alpha)(1+\beta)(2-\beta)C_{i+1,j-1}^{k} \\
& - \tfrac{1}{12}\alpha(1-\alpha^2)(2-\beta)(1-\beta^2)C_{i-2,j}^{k} + \tfrac{1}{4}\alpha(1+\alpha)(2-\alpha)(2-\beta)(1-\beta^2)C_{i-1,j}^{k}
\end{aligned}
$$

$$+\tfrac{1}{4}(2-\alpha)(1-\alpha^2)(2-\beta)(1-\beta^2)C^k_{i,j}-\tfrac{1}{12}\alpha(1-\alpha)(2-\alpha)(2-\beta)(1-\beta^2)C^k_{i+1,j}$$
$$+\tfrac{1}{36}\alpha\beta(1-\alpha^2)(1-\beta)(2-\beta)C^k_{i-2,j+1}-\tfrac{1}{12}\alpha\beta(1+\alpha)(2-\alpha)(1-\beta)(2-\beta)C^k_{i-1,j+1}$$
$$-\tfrac{1}{12}\beta(2-\alpha)(1-\alpha^2)(1-\beta)(2-\beta)C^k_{i,j+1}+\tfrac{1}{36}\alpha\beta(1-\alpha)(2-\alpha)(1-\beta)(2-\beta)C^k_{i+1,j+1}$$

A.2. TOP-12

Retaining terms of up to fourth order in polynomial (18), using computational and flux stencils as shown in Figure 7, and requiring that the polynomial passes through all nodes from the computational stencil, one obtains the following system of linear equations:

$$\begin{vmatrix}
1 & 1 & 1 & 1 & 2 & 2 & 2 & 2 & 4 & 4 & 8 & 8\\
1 & 0 & 0 & 0 & 2 & 0 & 0 & 0 & 4 & 0 & 8 & 0\\
1 & 2 & 4 & 8 & 1 & 2 & 4 & 8 & 1 & 2 & 1 & 2\\
1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1\\
1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0\\
1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1 & 1 & -1\\
1 & 2 & 4 & 8 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\
1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\
1 & -1 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0\\
1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 & 1 & 1 & -1 & -1\\
1 & 0 & 0 & 0 & -1 & 0 & 0 & 0 & 1 & 0 & -1 & 0
\end{vmatrix}
\begin{vmatrix} f_{0,0}\\ f_{1,0}\\ f_{2,0}\\ f_{3,0}\\ f_{0,1}\\ f_{1,1}\\ f_{2,1}\\ f_{3,1}\\ f_{0,2}\\ f_{1,2}\\ f_{0,3}\\ f_{1,3} \end{vmatrix}
=
\begin{vmatrix} C^k_{i-1,j-2}\\ C^k_{i,j-2}\\ C^k_{i-2,j-1}\\ C^k_{i-1,j-1}\\ C^k_{i,j-1}\\ C^k_{i+1,j-1}\\ C^k_{i-2,j}\\ C^k_{i-1,j}\\ C^k_{i,j}\\ C^k_{i+1,j}\\ C^k_{i-1,j+1}\\ C^k_{i,j+1} \end{vmatrix}$$

which also can be rewritten in the matrix form as Equation (6). The inverse matrix $\mathbf{B}^{-1}$ is found using the Gauss–Jordan method, as

$$\mathbf{B}^{-1}=\begin{vmatrix}
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0\\
0 & 0 & 0 & 0 & 0 & 0 & -1/6 & 1 & -1/2 & -1/3 & 0 & 0\\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 1/2 & -1 & 1/2 & 0 & 0\\
0 & 0 & 0 & 0 & 0 & 0 & 1/6 & -1/2 & 1/2 & -1/6 & 0 & 0\\
0 & -1/6 & 0 & 0 & 1 & 0 & 0 & 0 & -1/2 & 0 & 0 & -1/3\\
-1/6 & 1/6 & -1/6 & 1 & -1/2 & -1/3 & 1/6 & -1/2 & 0 & 1/3 & -1/3 & 1/3\\
0 & 0 & 0 & 1/2 & -1 & 1/2 & 0 & -1/2 & 1 & -1/2 & 0 & 0\\
0 & 0 & 1/6 & -1/2 & 1/2 & -1/6 & -1/6 & 1/2 & -1/2 & 1/6 & 0 & 0\\
0 & 0 & 0 & 0 & 1/2 & 0 & 0 & 0 & -1 & 0 & 0 & 1/2\\
0 & 0 & 0 & 1/2 & -1/2 & 0 & 0 & -1 & 1 & 0 & 1/2 & -1/2\\
0 & 1/6 & 0 & 0 & -1/2 & 0 & 0 & 0 & 1/2 & 0 & 0 & -1/6\\
1/6 & -1/6 & 0 & -1/2 & 1/2 & 0 & 0 & 1/2 & -1/2 & 0 & -1/6 & 1/6
\end{vmatrix}$$

Using relationship (7), the computed coefficients $\mathbf{f}$ have to be substituted into polynomial (5) to obtain the single-step update as in (4), where

$$\bar{\mathbf{S}} = (C^k_{i-1,j-2}, C^k_{i,j-2}, C^k_{i-2,j-1}, C^k_{i-1,j-1}, C^k_{i,j-1}, C^k_{i+1,j-1}, C^k_{i-2,j}, C^k_{i-1,j}, C^k_{i,j}$$
$$C^k_{i+1,j}, C^k_{i-1,j+1}, C^k_{i,j+1})$$

$$p_1 = -\tfrac{1}{6}\alpha\beta(1-\beta^2), \quad p_2 = -\tfrac{1}{6}\beta(1-\alpha)(1-\beta^2), \quad p_3 = -\tfrac{1}{6}\alpha\beta(1-\alpha^2)$$

$$p_4 = \tfrac{1}{2}\alpha\beta(2+\alpha-\alpha^2+\beta-\beta^2), \quad p_5 = \tfrac{1}{2}\beta(1-\alpha)(2+\alpha-\alpha^2+\beta-\beta^2)$$

$$p_6 = -\tfrac{1}{6}\alpha\beta(1-\alpha)(2-\alpha), \quad p_7 = -\tfrac{1}{6}\alpha(1-\alpha^2)(1-\beta)$$

$$p_8 = \tfrac{1}{2}\alpha(1-\beta)(2+\alpha-\alpha^2+\beta-\beta^2), \quad p_9 = \tfrac{1}{2}(1-\alpha)(1-\beta)(2+\alpha-\alpha^2+\beta-\beta^2)$$

$$p_{10} = -\tfrac{1}{6}\alpha(1-\alpha)(2-\alpha)(1-\beta), \quad p_{11} = -\tfrac{1}{6}\alpha\beta(1-\beta)(2-\beta)$$

$$p_{12} = -\tfrac{1}{6}\beta(1-\alpha)(1-\beta)(2-\beta)$$

Second derivatives of the polynomial P with respect to α and β are

$$\frac{\partial^2 P}{\partial\alpha^2} = \alpha\beta C^k_{i-2,j-1} + (1-3\alpha)\beta C^k_{i-1,j-1} - (2-3\alpha)\beta C^k_{i,j-1}$$
$$+ (1-\alpha)\beta C^k_{i+1,j-1} + \alpha(1-\beta)C^k_{i-2,j} + (1-3\alpha)(1-\beta)C^k_{i-1,j}$$
$$- (2-3\alpha)(1-\beta)C^k_{i,j} + (1-\alpha)(1-\beta)C^k_{i+1,j}$$

$$\frac{\partial^2 P}{\partial\beta^2} = \alpha\beta C_{i-1,j-2} + \alpha(1-3\beta)C_{i-1,j-1} - \alpha(2-3\beta)\beta C_{i-1,j}$$
$$+ \alpha(1-\beta)C_{i-1,j+1} + (1-\alpha)\beta C_{i,j-2} + (1-\alpha)(1-3\beta)C_{i,j-1}$$
$$- (1-\alpha)(2-3\beta)C_{i,j} + (1-\alpha)(1-\beta)C_{i,j+1}$$

The amplification factor for the TOP-12 scheme reads

$$G = p_6 + p_9 + p_{11} + (p_5 + p_8 + p_{10} + p_{12})c_1 + (p_2 + p_7)c_2 + p_4(c_1^2 - s_2^2)$$
$$+ (p_1 + p_3)(c_1c_2 - s_1s_2) + I[-(p_5 + p_8 - p_{10} - p_{12})s_1 - (p_2 + p_7)s_2$$
$$- (p_1 + p_3)(c_1s_2 - s_1c_2) - 2p_4c_1s_1]$$

where $c_1 = \cos(\theta), s_1 = \sin(\theta), c_2 = \cos(2\theta), \ s_2 = \sin(2\theta)$.

A.3. TOP-10A

Coefficients of polynomial (5) for the TOP-10A scheme (Figure 13) are defined by the system of linear equations

$$\begin{vmatrix} 1 & 1 & 1 & 1 & -1 & -1 & -1 & 1 & 1 & -1 \\ 1 & 0 & 0 & 0 & 2 & 0 & 0 & 4 & 0 & 8 \\ 1 & -1 & 1 & -1 & 1 & -1 & 1 & 1 & -1 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 1 & 2 & 4 & 8 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & -1 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & -1 & 0 & 0 & 1 & 0 & -1 \end{vmatrix} \begin{vmatrix} f_{0,0} \\ f_{1,0} \\ f_{2,0} \\ f_{3,0} \\ f_{0,1} \\ f_{1,1} \\ f_{2,1} \\ f_{0,2} \\ f_{1,2} \\ f_{0,3} \end{vmatrix} = \begin{vmatrix} C^k_{i-1,j+1} \\ C^k_{i,j-2} \\ C^k_{i+1,j-1} \\ C^k_{i-1,j-1} \\ C^k_{i,j-1} \\ C^k_{i-2,j} \\ C^k_{i-1,j} \\ C^k_{i,j} \\ C^k_{i+1,j} \\ C^k_{i,j+1} \end{vmatrix}$$

leading to the following inverse matrix:

$$\mathbf{B}^{-1} = \begin{vmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1/6 & 1 & -1/2 & -1/3 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1/2 & -1 & 1/2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1/6 & -1/2 & 1/2 & -1/6 & 0 \\ 0 & -1/6 & 0 & 0 & 1 & 0 & 0 & -1/2 & 0 & -1/3 \\ -1/2 & 0 & -1/2 & 0 & 1/2 & 0 & 1/2 & -1 & 1/2 & 1/2 \\ 0 & 0 & 1/2 & 1/2 & -1 & 0 & -1/2 & 1 & -1/2 & 0 \\ 0 & 0 & 0 & 0 & 1/2 & 0 & 0 & -1 & 0 & 1/2 \\ 1/2 & 0 & 0 & 1/2 & -1/2 & 0 & -1 & 1 & 0 & -1/2 \\ 0 & 1/6 & 0 & 0 & -1/2 & 0 & 0 & 1/2 & 0 & -1/6 \end{vmatrix}$$

The resulting single-step update for the TOP-10A convective scheme reads

$$\begin{aligned} C^{k+1}_{i,j} = & -\tfrac{1}{2}\alpha\beta(1-\beta)C^k_{i-1,j+1} - \tfrac{1}{6}\beta(1-\beta^2)C^k_{i,j-2} - \tfrac{1}{2}\alpha\beta(1-\alpha)C^k_{i+1,j-1} \\ & + \tfrac{1}{2}\alpha\beta(\alpha+\beta)C^k_{i-1,j-1} + \tfrac{1}{2}\beta((1+\alpha)(2-\alpha)+\beta(1-\alpha-\beta))C^k_{i,j-1} \\ & - \tfrac{1}{6}\alpha(1-\alpha^2)C^k_{i-2,j} + \tfrac{1}{2}\alpha((1+\beta)(2-\beta)+\alpha(1-\alpha-\beta))C^k_{i-1,j} \\ & + \tfrac{1}{2}((1-\alpha^2)(2-\alpha)+(1-\beta^2)(2-\beta)-2\alpha\beta(1-\alpha-\beta)-2)C^k_{i,j} \\ & - \tfrac{1}{6}\alpha(1-\alpha)(2-\alpha-3\beta)C^k_{i+1,j} - \tfrac{1}{6}\beta(1-\beta)(2-3\alpha-\beta)C^k_{i,j+1} \end{aligned}$$

A.4. TOP-10B

For the stencil shown in Figure 14, the polynomial coefficients are defined by the set of linear equations

$$\begin{vmatrix} 1 & 1 & 1 & 1 & 2 & 2 & 2 & 4 & 4 & 8 \\ 1 & 0 & 0 & 0 & 2 & 0 & 0 & 4 & 0 & 8 \\ 1 & 2 & 4 & 8 & 1 & 2 & 4 & 1 & 2 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 1 \\ 1 & 2 & 4 & 8 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & -1 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & -1 & 0 & 0 & 1 & 0 & -1 \end{vmatrix} \begin{vmatrix} f_{0,0} \\ f_{1,0} \\ f_{2,0} \\ f_{3,0} \\ f_{0,1} \\ f_{1,1} \\ f_{2,1} \\ f_{0,2} \\ f_{1,2} \\ f_{0,3} \end{vmatrix} = \begin{vmatrix} C^k_{i-1,j-2} \\ C^k_{i,j-2} \\ C^k_{i-2,j-1} \\ C^k_{i-1,j-1} \\ C^k_{i,j-1} \\ C^k_{i-2,j} \\ C^k_{i-1,j} \\ C^k_{i,j} \\ C^k_{i+1,j} \\ C^k_{i,j+1} \end{vmatrix}$$

which leads to the inverse matrix

$$\mathbf{B}^{-1} = \begin{vmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1/6 & 1 & -1/2 & -1/3 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1/2 & -1 & 1/2 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1/6 & -1/2 & 1/2 & -1/6 & 0 \\ 0 & -1/6 & 0 & 0 & 1 & 0 & 0 & -1/2 & 0 & -1/3 \\ -1/2 & 1/2 & -1/2 & 3 & -2.5 & 1/2 & -2.5 & 2 & 0 & 0 \\ 0 & 0 & 1/2 & -1 & 1/2 & -1/2 & 1 & -1/2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1/2 & 0 & 0 & -1 & 0 & 1/2 \\ 1/2 & -1/2 & 0 & -1 & 1 & 0 & 1/2 & -1/2 & 0 & 0 \\ 0 & 1/6 & 0 & 0 & -1/2 & 0 & 0 & 1/2 & 0 & -1/6 \end{vmatrix}$$

Single-step update for the TOP-10B convective scheme becomes

$$\begin{aligned} C^{k+1}_{i,j} = & -\tfrac{1}{2}\alpha\beta(1-\beta)C^k_{i-1,j-2} - \tfrac{1}{6}\beta(1-\beta)(1+\beta-3\alpha)C^k_{i,j-2} \\ & - \tfrac{1}{2}\alpha\beta(1-\alpha)C^k_{i-2,j-1} + \alpha\beta(3-\alpha-\beta)C^k_{i-1,j-1} - \tfrac{1}{6}\alpha(1-\alpha)(2-\alpha)C^k_{i+1,j} \\ & + \tfrac{1}{2}\beta((1+\beta)(2-\beta) - \alpha(5-\alpha-2\beta))C^k_{i,j-1} - \tfrac{1}{6}\alpha(1-\alpha)(1+\alpha-3\beta)C^k_{i-2,j} \\ & + \tfrac{1}{2}\alpha((1+\alpha)(2-\alpha) - \beta(5-2\alpha-\beta))C^k_{i-1,j} - \tfrac{1}{6}\beta(1-\beta)(2-\beta)C^k_{i,j+1} \\ & + \tfrac{1}{2}((1-\alpha^2)(2-\alpha) + (1-\beta^2)(2-\beta) + \alpha\beta(4-\alpha-\beta) - 2)C^k_{i,j} \end{aligned}$$

ACKNOWLEDGEMENTS

The authors express their gratitude to the anonymous reviewers for their productive comments during the paper reviewing.

REFERENCES

1. Leonard BP. Third-order upwinding as a rational basis for computational fluid dynamics. In *Proceedings of the International Computational Techniques and Applications Conference* (CTAC-83), The University of Sydney, Australia, 1983. Elsevier: Amsterdam, 1984; 106–120.
2. Leonard BP. A stable and accurate convective modelling procedure based on quadratic upstream interpolation. *Computer Methods in Applied Mechanics and Engineering* 1979; **19**:59–98.
3. Davis RW, Moore EF. A numerical study of vortex shedding from rectangles. *Journal of Fluid Mechanics* 1982; **116**:475–506.
4. Chen Y, Falconer R. Advection–diffusion modelling using the modified QUICK scheme. *International Journal for Numerical Methods in Fluids* 1992; **15**:1171–1196.
5. Chen Y, Falconer R. Modified forms of the third-order convection, second-order diffusion scheme for the advection–diffusion equation. *Advances in Water Resources* 1994; **17**:147–170.
6. Leonard BP, Niknafs H. Cost-effective accurate coarse-grid method for highly convective multidimensional unsteady flow. In *Proceedings of the CFD Symposium on Aeropropulsion*, NASA Lewis Research Center, Cleveland, OH, USA, April 1990; 227–240.
7. Rasch PJ. Conservative shape-preserving two-dimensional transport on a spherical reduced grid. *Monthly Weather Review* 1994; **122**:1337–1350.
8. Ekebjærg L, Justesen P. An explicit scheme for advection–diffusion modelling in two dimensions. *Computer Methods in Applied Mechanics and Engineering* 1991; **88**:287–297.
9. LeVeque RJ. High-resolution conservative algorithms for advection in incompressible flow. *SIAM Journal of Numerical Analysis* 1996; **33**:627–665.
10. Leonard BP, MacVean MK, Lock AP. The flux integral method for multidimensional convection and diffusion. *Applied Mathematical Modelling* 1995; **19**:333–342.
11. Balzano A. MOSQUITO: an efficient finite difference scheme for numerical simulation of 2D advection. *International Journal for Numerical Methods in Fluids* 1999; **31**:481–496.
12. Gerges H, McCorquodale JA. Modeling of flow in rectangular sedimentation tanks by an explicit third-order upwinding technique. *International Journal for Numerical Methods in Fluids* 1997; **24**:537–561.
13. Leonard BP. Note on the von Neumann stability of explicit one-dimensional advection schemes. *Computer Methods in Applied Mechanics and Engineering* 1994; **118**:29–46.
14. Leonard BP. Order of accuracy of QUICK and related convection–diffusion schemes. *Applied Mathematical Modelling* 1995; **19**:640–653.

