

Hopping numerical approximations of the hyperbolic equation

Pavel Tkalich^{*,†}

*Tropical Marine Science Institute, National University of Singapore, 14 Kent Ridge Rd.,
Singapore 119223, Singapore*

SUMMARY

Polynomial functions can be used to derive numerical schemes for an approximate solution of hyperbolic equations. A conventional derivation technique requires a polynomial to pass through every node values of a continuous computational stencil, leading to severe manifestation of the Gibbs phenomenon and strict time-step limitation. To overcome the problem, this paper introduces polynomials that skip regularly ('hop' over) one or more nodes from the computational grid. Polynomials hopping over odd and even nodes yield a series of explicit numerical schemes of a required accuracy, with Lax–Friedrichs method being a particular simplest case. The schemes have two times wider stability interval compared to conventional continuous-stencil explicit methods. Convex combinations of odd- and even-node-based updates improve further accuracy and stability of the method. Out of considered combinations (up to third-order accuracy), derived odd-order methods are stable for the Courant number ranging from 0 to 3, and even-order ones from 0 to 5. A 2-D extension of the hopping polynomial method exhibits similar properties. Copyright © 2007 John Wiley & Sons, Ltd.

Received 24 February 2006; Revised 19 February 2007; Accepted 31 March 2007

KEY WORDS: high order; hyperbolic equation; hopping polynomials; large Courant number

1. INTRODUCTION

The quest for accurate and efficient numerical solutions of the hyperbolic conservation law

$$\frac{\partial \phi}{\partial t} + \frac{\partial u \phi}{\partial x} = 0 \quad (1)$$

is still far from completion. Here, $\phi(t, x)$ is the transported scalar, u is the uniform velocity positive along the increasing x space coordinate, t is the time. Equation (1) is considered as a simplified

*Correspondence to: Pavel Tkalich, Tropical Marine Science Institute, National University of Singapore, 14 Kent Ridge Rd., Singapore 119223, Singapore.

†E-mail: tmspt@nus.edu.sg

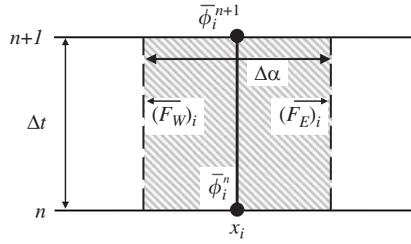


Figure 1. Sketch of a computational cell.

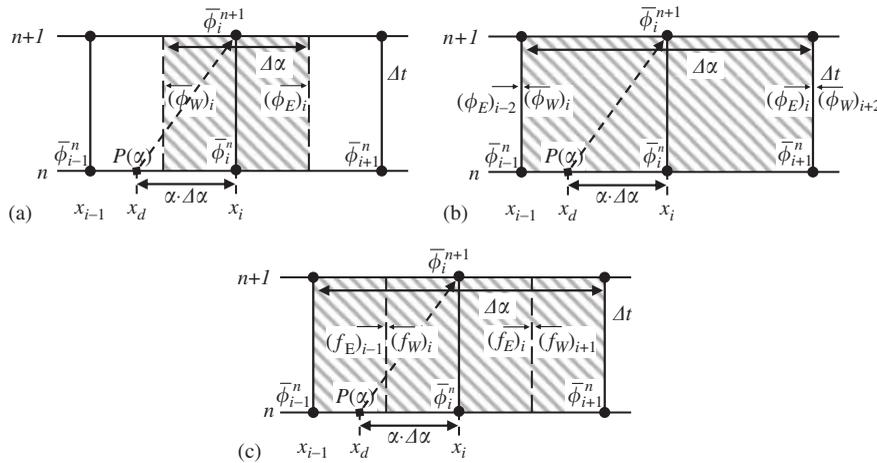


Figure 2. Sketch of computational cells having width: (a) $\Delta\alpha = \Delta x$, (b) and (c) $\Delta\alpha = 2\Delta x$.

model for a wide range of heat and mass transport problems in fluids. For a space–time grid (as in Figure 1), integration of Equation (1) over $\Delta\alpha$ and Δt gives the finite-volume formulation

$$\bar{\phi}_i^{n+1} = \bar{\phi}_i^n - (F_E)_i + (F_W)_i \tag{2}$$

Here, the bars indicate spatial averages over computational cell i at time levels n and $(n + 1)$, and east and west time-averaged fluxes have been introduced as $(F_E)_i = \alpha(\phi_E)_i$ and $(F_W)_i = \alpha(\phi_W)_i$; where $(\phi_E)_i$ and $(\phi_W)_i$ are the east and west time-averaged face values of the transported scalar, and α is the *computational Courant number*:

$$\alpha = u\Delta t/\Delta\alpha \tag{3}$$

where $\Delta t = t^{n+1} - t^n$ is the time step and $\Delta\alpha$ is the size of the *computational cell*.

Importantly, size of a computational cell $\Delta\alpha$ need not necessarily equal the *grid cell* size Δx , as illustrated in Figure 2. At a uniform grid in space-and-time domain, if the computational cell is a rectangle with the lower left corner at the point $(t^n, x_{i-1/2})$ and the upper right one at $(t^{n+1}, x_{i+1/2})$, as in Figure 2(a), then $\Delta\alpha = (x_{i+1/2} - x_{i-1/2}) = \Delta x$. If the computational cell consists of two grid cells as in Figure 2(b), then $\Delta\alpha = (x_{i+1} - x_{i-1}) = 2\Delta x$.

Conservation of mass by approximation (2) is guaranteed, if F_W of i th computational cell is equal to F_E of the adjacent one from the left. Problem (1) can be reformulated as to find (reconstruct) fluxes F_E , F_W , satisfying Equation (2) up to a required order of accuracy. There are several methods available to obtain high-order (higher than first) reconstructions; however, if special algorithmic precautions (the so-called flux limiters) are not applied, the approximations seem inevitably lead to spurious oscillations of the numerical solution. The limiters decrease an order of approximation in areas where oscillations might occur, and make resulting algorithms less efficient. Computational scientists spend decades to balance often contradicting requirements within a single scheme, but the problem is still far from being completely resolved. Unable to fit growing computational demands, the traditional approach to have a single flux expression, single computational stencil, or single grid is giving a way to more aggressive hybrid schemes with embedded automatic choice of the best flux, stencil, or grid. This trend is observed particularly in the development of ENO [1] and WENO [2] schemes. WENO scheme assigns each computational cell to several corresponding stencils and to a convexly weighted combination of the corresponding interpolating polynomials. Using computationally intensive logic, the weights are chosen to achieve the essentially non-oscillatory property that is important for shock wave computations. However, in many linear applications of heat and mass transport the sharp gradients are absent, and a focus of new schemes' development can be shifted to improvement of some other important properties, such as stability range. Thus, the idea to use a convex combination of schemes and stencils to improve efficiency of numerical algorithms has partially influenced this paper.

Another inspiration is given by Lax–Friedrichs (LxF) method [3]:

$$\bar{\phi}_i^{n+1} = \frac{1}{2}(\bar{\phi}_{i+1}^n + \bar{\phi}_{i-1}^n) - \frac{1}{2}(c\bar{\phi}_{i+1}^n - c\bar{\phi}_{i-1}^n) \quad (4)$$

The scheme can be derived by requiring that the first-degree polynomial $P_i^{(1)}(c) = g_i + h_i c$, defined at $[x_{i-1}, x_{i+1}]$, misses the central node value $\bar{\phi}_i^n$ while passing through values at both sides of the interval. Here, g_i and h_i are the coefficients to be determined; c is the local variable, corresponding to the *grid Courant number*

$$c = u\Delta t/\Delta x \quad (5)$$

Δx is the grid size. In terms of the *grid Courant number*, the LxF scheme is stable at $-1 \leq c \leq 1$ (*two grid cells*), which looks atypical for odd-order schemes known to be stable at a single grid cell (see extensive discussions of Leonard [4]). However, in terms of the *computational Courant number*, this is a single *computational cell* $\Delta x = 2\Delta x$ and the LxF scheme still holds the conventional single-computational-cell stability condition $|\alpha| \leq 1/2$, which follows from definition (3) and is illustrated in Figure 2(b). Two lessons can be learned from the basic LxF scheme: (1) the computational cell does not need to coincide with the grid-cell and (2) high-order LxF-type schemes can be derived using high-degree polynomials passing through every second node.

One can adjoin any number (l) of grid cells into a single computational cell as $\Delta x = l\Delta x$ and construct first-order LxF-type scheme at the extended stencil. In terms of grid- and computational-Courant numbers, the conventional (equal-or-less-than-a-unity) CFL condition extends to respective stability criteria

$$|c| \leq \frac{l}{2} \quad \text{or} \quad |\alpha| \leq \frac{1}{2} \quad (6)$$

The second condition in relationships (6) stems from definition (3) and the fact that LxF scheme is centred in space.

2. BASICS OF HOPPING APPROXIMATIONS

2.1. Suppression of the Gibbs phenomenon

For smooth solutions, the higher the order of the approximation, the better the accuracy; however, in the vicinity of discontinuities or shock fronts the Gibbs phenomenon may lead to spurious oscillations. By increasing the order of approximation one can reduce the ripples, but cannot get rid of them completely. Richards [5] shows that the overshoot tends to the typical 8.95% of shock magnitude as the degree of spline approximation approaches infinity. Almost all existing treatments for the Gibbs phenomenon reduction fall in the direction of summability (or averaging) methods, such as of Fejér [6] or Lanczos [7]. In particular, some suppression of oscillations can be achieved combining two or more periodic approximating functions having different phases. Figure 3 shows an example of ripples amplitude reduction by a factor of $\sqrt{2}$ after averaging two sine functions shifted $\pi/2$ relatively each other. Similar shift can be achieved numerically letting the functions to pass through alternative sets of odd and even nodes of a computational grid. One expects further improvement of the oscillation suppression by combining a larger number and variety of carefully selected functions.

Extending the idea to Lagrange polynomials, it is clear that a combination of two approximations, one utilizing even node and another odd ones, would require introduction of a computational cell consisting of two grid cells, as in Figure 2(b). In this case each of the polynomials ‘hops’ over nodes of the partner polynomial. Figure 4 illustrates approximation of a step function by two fifth-degree hopping polynomials (HOPs), $P_1^{(5)}$ at stencil $\{x_0, x_2, \dots, x_{10}\}$ and $P_2^{(5)}$ at stencil $\{x_{-1}, x_1, \dots, x_9\}$, as compared with the conventional fifth-degree polynomial $P_5^{(5)}$ at $\{x_0, x_1, \dots, x_5\}$.

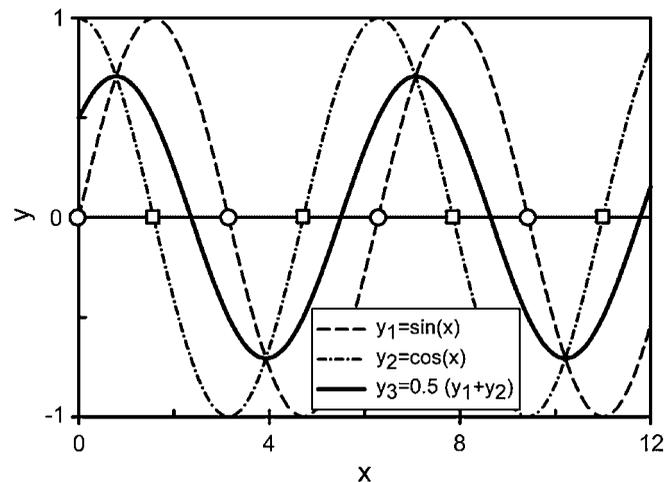


Figure 3. Sketch of the Gibbs phenomenon suppression by averaging two periodic curves passing through alternative grid-node sets (○-odd and □-even).

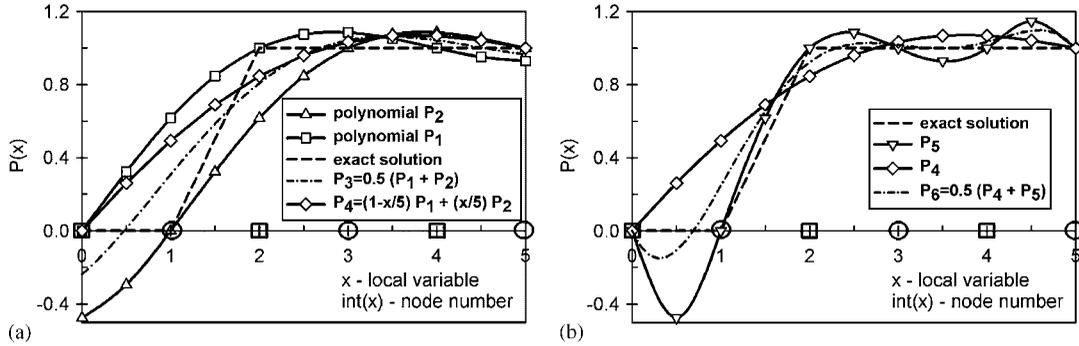


Figure 4. Square wave approximation using different combinations of hopping polynomials.

The step function is set to zero at nodes having indices less than or equal 1, and a unity otherwise; and an approximation error is compared at interval [0, 5] according to the measure

$$\varepsilon = \frac{\sum_i |\bar{\phi}_i - \phi_i^*|}{\sum_i |\phi_i^*|} \tag{7}$$

where ϕ^* is the exact solution, and $\bar{\phi}$ is the numerical one. Polynomial $P_1^{(5)}$ approximates the function with the error $\varepsilon_1 = 0.229$, which is comparable with that of the polynomial $P_2^{(5)}$ ($\varepsilon_2 = 0.214$). Polynomial $P_5^{(5)}$ has smaller error ($\varepsilon_5 = 0.168$), but exhibits more intensive wiggling. Average of two polynomials, $P_3^{(5)} = (P_1^{(5)} + P_2^{(5)})/2$, demonstrates a smooth approximation with smaller error $\varepsilon_3 = 0.123$. Convex combination $P_4^{(5)} = (1 - x/5)P_1^{(5)} + (x/5)P_2^{(5)}$ with the error $\varepsilon_4 = 0.178$ is less accurate than $P_3^{(5)}$ and $P_5^{(5)}$, but still much better as compared with $P_1^{(5)}$ and $P_2^{(5)}$ alone. Approximation $P_6^{(5)} = (P_4^{(5)} + P_5^{(5)})/2$ with the error $\varepsilon_6 = 0.099$ is an overall champion of the experiments. The trend is clear that a convex combination of polynomials is suppressing the Gibbs phenomenon, otherwise severe for each of the constituent polynomials.

2.2. Derivation of numerical schemes using polynomials

Algorithm (2) has to be completed by choosing one of available methods of flux reconstruction. In this paper, Lagrange polynomial at time level t^n is used to find a single-step update at time level t^{n+1} , from which the flux expression is recovered. Semi-Lagrangian technique yields an exact solution of Equation (1) over time Δt as (Figure 2)

$$\bar{\phi}_i^{n+1} = \phi(t^{n+1}, x_i) = \phi(t^n, x_d) \tag{8}$$

expressing the concept that transported scalar properties at the departure point (t^n, x_d) are advected to the arrival point (t^{n+1}, x_i) without changing. (Here and further down, some terminology of Leonard (2002) is adopted.) The transported scalar value at the departure point (t^n, x_d) can be approximated with the polynomial

$$\phi(t^n, x_d) = P^{(M)}(x) = \sum_{m=0}^M \alpha^m b_m \tag{9}$$

Here, M is the degree of the polynomial, and unknown coefficients $\mathbf{B} = \{b_m\}_{m=0, M}$ depend on values of the transported scalar ϕ at nodes of the computational stencil $\mathbf{X} = \{x_k\}_{k=0, M}$. To define uniquely coefficients \mathbf{B} using Lagrange polynomial elements, $M + 1$ linear equations $P(x_k) = \bar{\phi}_k^n$ can be formulated, requiring that the polynomial passes through every node of the computational stencil \mathbf{X} , i.e.

$$\sum_{m=0}^M a_{mk} b_m = \bar{\phi}_k^n \quad (10)$$

A matrix form of Equations (10) is $\mathbf{AB} = \mathbf{\Phi}$, where components of matrix $\mathbf{A} = \{a_{mk}\}$ and vector $\mathbf{\Phi} = \{\bar{\phi}_k^n\}$ are known, and vector \mathbf{B} has to be identified. Solution of the system can be obtained by finding the inverse matrix \mathbf{A}^{-1} , and then $\mathbf{B} = \mathbf{A}^{-1}\mathbf{\Phi}$. Finally, combining Equations (8) and (9), and substituting found coefficients $\{b_m\}$, one obtains explicit single-step update

$$\bar{\phi}_i^{n+1} = P^{(M)}(\alpha) \quad (11)$$

To match conservative form (2), the east-face flux can be derived equalizing right-hand sides of Equations (2) and (11), leading to

$$(F_E)_i = \bar{\phi}_i^n - P^{(M)}(\alpha) + (F_E)_{\text{west neighbor cell}} \quad (12)$$

Using HOPs of degree up to third, a series of schemes is deduced in the paper. The derivation is limited to polynomials hopping over a single node only; even though the method allows skipping any number of nodes. For a convenience of reference all schemes are identified by the upper index ' kHm ', where ' k ' refers to the order of approximation, ' m ' is to quantify a stencil bias in upwind direction relatively the arrival point x_i , and ' H ' indicates usage of HOPs in the derivation. For a consistency with previous findings, the order of accuracy of the schemes developed in this paper is given in terms of $\Delta\alpha$, if otherwise is not stated. Following the tendency of conventional passing-through-each-grid-node polynomials, it is expected that an M th degree HOP produces M th-order (in $\Delta\alpha$) accuracy scheme with the truncation error

$$e \sim 0(\Delta\alpha^M) \sim 0(l^M \Delta x^M) \quad (13)$$

Relationship (13) indicates that the error increases with a number of grid cells (l) included in the in a single computational cell.

A series of numerical tests is performed to compare the performance of the schemes. In one of the tests, an idealized 1-D computational domain consists of 10 000 nodes, numbered from left to right, with the grid size $\Delta x = 100$ m. A uniform velocity $u = 1$ m/s is set in the entire domain. The initial zero values are set in the domain, except for the interval $(150\Delta x, 250\Delta x)$, where $\bar{\phi}^0 = 1$. It is expected that for a pure advection an ideal numerical scheme does not introduce a distortion into the initial shape, while moving it with the flow. Computations are conducted to allow the initial profile to advect to the opposite side of the domain, where numerical solutions are compared with the exact one.

3. HIERARCHY OF HOPPING SCHEMES

Since, a computational stencil may skip each second node, two cases are possible: (a) the arrival point x_i is omitted from the departure stencil, such as $\{\dots, x_{i-3}, x_{i-1}, x_{i+1}, \dots\}$; and (b) the arrival point belongs to the departure stencil, as $\{\dots, x_{i-2}, x_i, x_{i+2}, \dots\}$.

In the first case (a) the conservative form (2) leads to

$$\bar{\phi}_i^{n+1} = \tilde{\phi}_i^n - (F_E)_i + (F_W)_i \tag{14a}$$

where the capital letter F identifies the double-grid-cell (DGC) fluxes, as illustrated in Figures 1 and 2(b); mass conservation is guaranteed if $(F_W)_i = (F_E)_{i-2}$; and $\tilde{\phi}_i^n$ approximates value of the transported scalar at point (t^n, x_i) .

In the second case (b), conservative form (2) simplifies to

$$\bar{\phi}_i^{n+1} = \bar{\phi}_i^n - (F_E)_i + (F_W)_i, \quad (F_W)_i = (F_E)_{i-2} \tag{14b}$$

For computational efficiency, Equations (14) can be rewritten uniquely in terms of single-grid-cell (SGC) pseudo-fluxes f_E and f_W (see Figure 2(c)) in order to utilize the conservative finite-volume form

$$\bar{\phi}_i^{n+1} = \bar{\phi}_i^n - (f_E)_i + (f_W)_i, \quad (f_W)_i = (f_E)_{i-1} \tag{15}$$

where

$$(f_E)_i = (\tilde{f}_E)_i + (F_E)_i + (F_E)_{i-1} \tag{16}$$

Functions, \tilde{f}_E and \tilde{f}_W , can be identified knowing that

$$\tilde{\phi}_i^n = \bar{\phi}_i^n - (\tilde{f}_E)_i + (\tilde{f}_W)_i \quad \text{and} \quad (\tilde{f}_W)_i = (\tilde{f}_E)_{i-1} \tag{17a}$$

In the particular case of (14b), relationships (17a) are simplified to

$$\tilde{\phi}_i^n = \bar{\phi}_i^n \quad \text{and} \quad (\tilde{f}_W)_i = (\tilde{f}_E)_i = 0 \tag{17b}$$

In this paper, the schemes are derived only for positive velocities u ; however, the algorithms can be rewritten straightforwardly for the opposite direction. Derived stencils are summarized in Figure 5, and stability regions are analysed graphically in Figure 6, where values of amplification

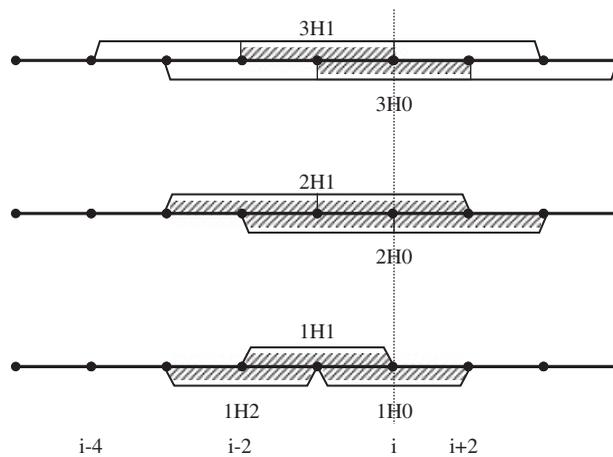


Figure 5. Stencils for some HOP schemes.

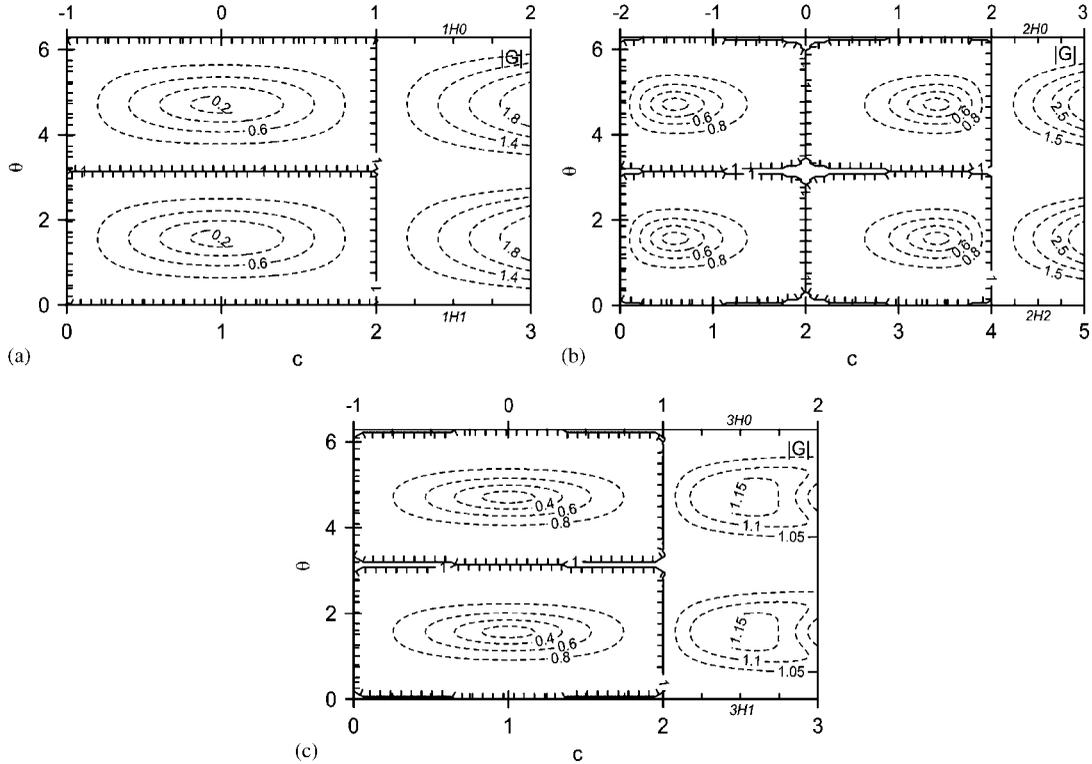


Figure 6. Amplification factor modulus $|G|$ for some HOP schemes: (a) 1H0 (LxF) and 1H1; (b) 2H0 and 2H2; and (c) 3H0 and 3H1. θ is the phase angle.

factor modulus less-than-a-unity are encircled by hatched curves. Here the von Neumann stability analysis is utilized, where solution of Equation (15) is sought in the form $\bar{\phi}_i^n = g^n \exp(i\theta)$, where $\theta = K\Delta x$ is the phase angle, g^n is the amplitude of the Fourier component at $n\Delta t$, K is the wave number; $I = \sqrt{-1}$. A difference scheme is considered stable for a chosen value of the Courant number c , if a modulus of amplification factor $|G(\theta, c)| = |g^{n+1}/g^n|$ is less or equal a unity for all phase angles over one complete time step.

For the first-degree polynomial ($M = 1$ in Equation (9)) defined at two nodes $\{x_{i-1}, x_{i+1}\}$ the first-order central in space single-step update

$$(\bar{\phi}_i^{n+1})^{1H0} = \frac{1}{2}[(1 - c)\bar{\phi}_{i+1}^n + (1 + c)\bar{\phi}_{i-1}^n] \tag{18}$$

is recognized as LxF scheme (4). One can obtain the DGC form (14a) using relationships $(\bar{\phi}_i^n)^{1H0} = (\bar{\phi}_{i+1}^n + \bar{\phi}_{i-1}^n)/2$ and $(f_E)_i^{1H0} = c\bar{\phi}_{i+1}^n/2$, or SGC form (15) using expression for the east-face pseudo-flux

$$(f_E)_i^{1H0} = -\frac{1}{2}(\bar{\phi}_{i+1}^n - \bar{\phi}_i^n) + \frac{1}{2}c(\bar{\phi}_{i+1}^n + \bar{\phi}_i^n) \tag{19}$$

Estimation (13) for $l = 2$ and $M = 1$ suggests that LxF scheme is two times more diffusive than a first order in Δx scheme. Indeed, for the first-order upwind (1UP) scheme

$$(\bar{\phi}_i^{n+1})^{1UP} = \bar{\phi}_i^n - c(\bar{\phi}_i^n - \bar{\phi}_{i-1}^n) \tag{20}$$

the numerical viscosity is derived as

$$(v)^{1UP} = \frac{u}{2}(1 - c)\Delta x$$

whereas LxF scheme (18) has the viscosity

$$(v)^{1H0} = \frac{u}{2c}(1 - c^2)\Delta x$$

Ratio of numerical viscosities of LxF and 1UP methods $(v)^{1H0}/(v)^{1UP} = (1 + c)/c$ tends to 2 when the Courant number tends to a unity, i.e. estimation (13) is valid.

Defining first-degree polynomial (9) at nodes $\{x_{i-2}, x_i\}$, the first-order upwind update follows

$$(\bar{\phi}_i^{n+1})^{1H1} = \frac{1}{2}[(2 - c)\bar{\phi}_i^n + c\bar{\phi}_{i-2}^n] \tag{21}$$

The scheme is stable at the interval $[x_{i-2}, x_i]$, leading to the stability condition $0 \leq c \leq 2$ (lower horizontal axis in Figure 6(a)). Due to the fact that the arrival point x_i belongs to the departure stencil $\{x_{i-2}, x_i\}$, relationships (14b) and (17b) are valid, and DGC and SGC fluxes become, respectively,

$$(F_E)_i^{1H1} = \frac{1}{2}c\bar{\phi}_i^n \quad \text{and} \quad (f_E)_i^{1H1} = \frac{1}{2}(c\bar{\phi}_i^n + c\bar{\phi}_{i-1}^n) \tag{22}$$

Shifting the computational stencil further and further upwind relatively the arrival node, one obtains sequentially conservative formulations (14a) or (14b), which can be rewritten in SGC form (15) using expressions (16) with definitions (17a) or (17b), respectively. Thus, the up-upwind first-order update at the interval $[x_{i-3}, x_{i-1}]$ is

$$(\bar{\phi}_i^{n+1})^{1H2} = \frac{1}{2}[(3 - c)\bar{\phi}_{i-1}^n - (1 - c)\bar{\phi}_{i-3}^n] \tag{23}$$

One obtains the conservative form of 1H2 method substituting expressions

$$(\tilde{f}_E)_i^{1H2} = \bar{\phi}_i^n - \frac{1}{2}(\bar{\phi}_{i-1}^n + \bar{\phi}_{i-2}^n) \quad \text{and} \quad (F_E)_i^{1H2} = \frac{1}{2}c\bar{\phi}_{i-1}^n \tag{24}$$

into relationships (15) and (16). The scheme is conditionally stable at $1 \leq c \leq 3$.

Second-degree polynomial (9) yields a series of second-order schemes. For the stencil $\{x_{i-2}, x_i, x_{i+2}\}$, the central in space single-step update follows

$$(\bar{\phi}_i^{n+1})^{2H0} = \frac{1}{8}[-c(2 - c)\bar{\phi}_{i+2}^n + 2(4 - c^2)\bar{\phi}_i^n + c(2 + c)\bar{\phi}_{i-2}^n] \tag{25}$$

Alternatively, DGC east-face flux

$$(F_E)_i^{2H0} = \frac{1}{8}c[(2 - c)\bar{\phi}_{i+2}^n + (2 + c)\bar{\phi}_i^n] \tag{26}$$

and expression $(\tilde{f}_E)_i^{2H0} = 0$ are useful for the conservative formulations (14b), (15) and (16). Scheme (25) is stable at $|c| \leq 2$ (four grid cells) as shown by the upper horizontal axis in Figure 6(b).

Defining interpolant at stencil $\{x_{i-3}, x_{i-1}, x_{i+1}\}$, one obtains second-order upwind update

$$(\bar{\phi}_i^{n+1})^{2H1} = \frac{1}{8}[(1-c)(3-c)\bar{\phi}_{i+1}^n + 2(1+c)(3-c)\bar{\phi}_{i-1}^n - (1-c^2)\bar{\phi}_{i-3}^n] \quad (27)$$

and respective DGC and SGC fluxes can be recovered using Equations (16), (17a) and

$$\begin{aligned} (\tilde{f}_E)_i^{2H1} &= \frac{1}{8}(-3\bar{\phi}_{i+1}^n + 5\bar{\phi}_i^n - \bar{\phi}_{i-1}^n - \bar{\phi}_{i-2}^n) \\ (F_E)_i^{2H1} &= \frac{1}{8}c[(4-c)\bar{\phi}_{i+1}^n + c\bar{\phi}_{i-1}^n] \end{aligned} \quad (28)$$

Here, (as always) west-face values are easily identified as $(\tilde{f}_W)_i = (\tilde{f}_E)_{i-1}$ and $(F_W)_i = (F_E)_{i-2}$ in order to utilize conservative formulations (14) and (15). Scheme (27) is stable at four grid cells ($-1 \leq c \leq 3$). More upwind schemes are derived by shifting the stencil further upwind relatively the arrival point x_i , to positions $\{x_{i-4}, x_{i-2}, x_i\}$ and $\{x_{i-5}, x_{i-3}, x_{i-1}\}$. In the first case, the single-step update and fluxes are given by

$$(\bar{\phi}_i^{n+1})^{2H2} = \frac{1}{8}[(2-c)(4-c)\bar{\phi}_i^n + 2c(4-c)\bar{\phi}_{i-2}^n - c(2-c)\bar{\phi}_{i-4}^n] \quad (29)$$

and

$$(\tilde{f}_E)_i^{2H2} = 0, \quad (F_E)_i^{2H2} = \frac{1}{8}c[(6-c)\bar{\phi}_i^n - (2-c)\bar{\phi}_{i-2}^n] \quad (30)$$

In the second case, these are

$$(\bar{\phi}_i^{n+1})^{2H3} = \frac{1}{8}[(3-c)(5-c)\bar{\phi}_{i-1}^n - 2(1-c)(5-c)\bar{\phi}_{i-3}^n + (1-c)(3-c)\bar{\phi}_{i-5}^n] \quad (31)$$

and

$$\begin{aligned} (\tilde{f}_E)_i^{2H3} &= \frac{1}{8}(8\bar{\phi}_i^n - 7\bar{\phi}_{i-1}^n - 7\bar{\phi}_{i-2}^n + 3\bar{\phi}_{i-3}^n + 3\bar{\phi}_{i-4}^n) \\ (F_E)_i^{2H3} &= \frac{1}{8}c[(8-c)\bar{\phi}_{i-1}^n - (4-c)\bar{\phi}_{i-3}^n] \end{aligned} \quad (32)$$

Schemes (29) and (31) are stable at $[x_{i-4}, x_i]$ and $[x_{i-5}, x_{i-1}]$, respectively, yielding the stability conditions $0 \leq c \leq 4$ and $1 \leq c \leq 5$.

Third-degree polynomial at the stencil $\{x_{i-3}, x_{i-1}, x_{i+1}, x_{i+3}\}$ leads to the third-order central update

$$\begin{aligned} (\bar{\phi}_i^{n+1})^{3H0} &= \frac{1}{48}[-(3-c)(1-c^2)\bar{\phi}_{i+3}^n + 3(1-c)(9-c^2)\bar{\phi}_{i+1}^n \\ &\quad + 3(1+c)(9-c^2)\bar{\phi}_{i-1}^n - (3+c)(1-c^2)\bar{\phi}_{i-3}^n] \end{aligned} \quad (33)$$

Fluxes for the conservative formulations (14) and (15) can be computed using relationships

$$\begin{aligned} (\tilde{f}_E)_i^{3H0} &= \frac{1}{16}(\bar{\phi}_{i+3}^n + \bar{\phi}_{i+2}^n - 8\bar{\phi}_{i+1}^n + 8\bar{\phi}_i^n - \bar{\phi}_{i-1}^n - \bar{\phi}_{i-2}^n) \\ (F_E)_i^{3H0} &= \frac{1}{48}c[-(1+3c-c^2)\bar{\phi}_{i+3}^n + 2(13-c^2)\bar{\phi}_{i+1}^n - (1-3c-c^2)\bar{\phi}_{i-1}^n] \end{aligned} \quad (34)$$

Recall that $(\tilde{f}_W)_i = (\tilde{f}_E)_{i-1}$ and $(F_W)_i = (F_E)_{i-2}$. The scheme is stable at the interval $[x_{i-1}, x_{i+1}]$, leading to the stability condition $|c| \leq 1$ (upper horizontal axis in Figure 6(c)).

Shifting computational stencil one node further upwind, another third-order single-step update follows

$$(\bar{\phi}_i^{n+1})^{3H1} = \frac{1}{48}[-c(2-c)(4-c)\bar{\phi}_{i+2}^n + 3(4-c^2)(4-c)\bar{\phi}_i^n + 3c(2+c)(4-c)\bar{\phi}_{i-2}^n - c(4-c^2)\bar{\phi}_{i-4}^n] \tag{35}$$

The respective east-face flux for DGC conservative formulation (14b) is

$$(F_E)_i^{3H1} = \frac{1}{48}c[(2-c)(4-c)\bar{\phi}_{i+2}^n + 2(2+c)(5-c)\bar{\phi}_i^n - (4-c^2)\bar{\phi}_{i-2}^n] \tag{36}$$

Alternatively, SGC expression (15) can be utilized knowing that $(\tilde{f}_E)_i^{3H1} = 0$. The scheme is stable at the interval $[x_{i-2}, x_i]$ ($0 \leq c \leq 2$), as shown by the lower horizontal axis in Figure 6(c).

Third-degree polynomial at the stencil $\{x_{i-5}, x_{i-3}, x_{i-1}, x_{i+1}\}$ yields update

$$(\bar{\phi}_i^{n+1})^{3H2} = \frac{1}{48}[(1-c)(3-c)(5-c)\bar{\phi}_{i+1}^n + 3(1+c)(3-c)(5-c)\bar{\phi}_{i-1}^n - 3(1-c^2)(5-c)\bar{\phi}_{i-3}^n + (1-c^2)(3-c)\bar{\phi}_{i-5}^n] \tag{37}$$

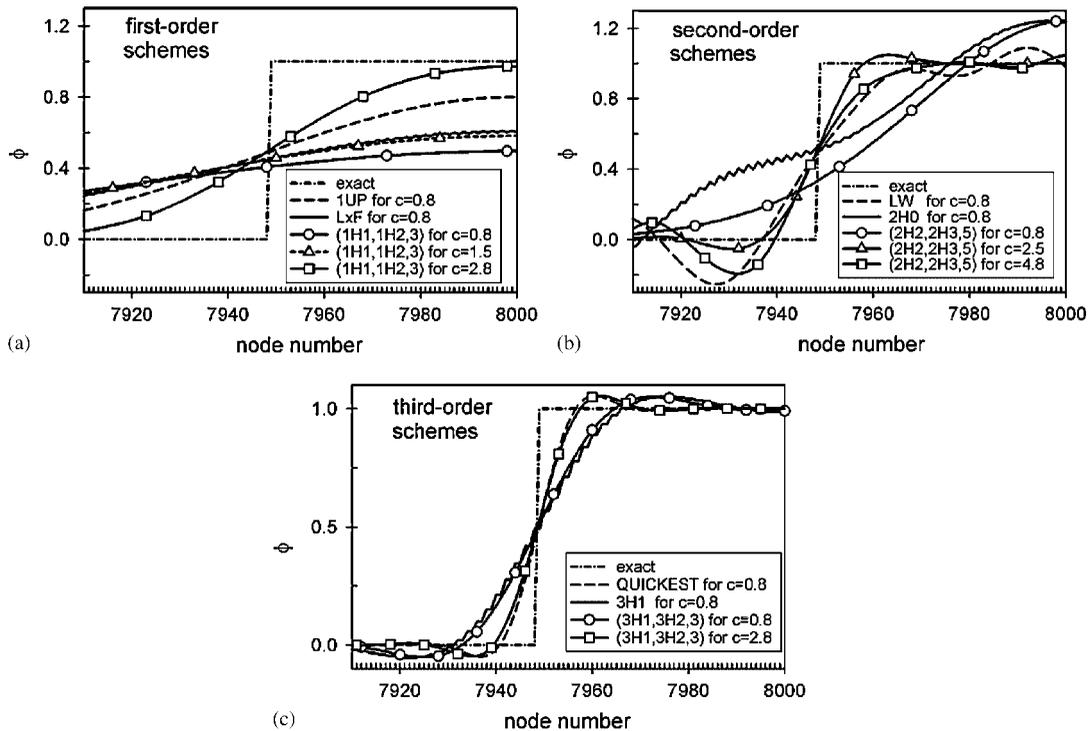


Figure 7. Performance of some HOP schemes using the square wave propagation test. (Only upwind half of the wave is shown.)

Table I. Error measure due to application of HOP schemes for the square wave propagation test.

First-order methods	Error ε	Second-order methods	Error ε	Third-order methods	Error ε
1UP, $c = 0.8$	0.621	LW, $c = 0.8$	0.281	QUICKEST, $c = 0.8$	0.082
1H0 (LxF), $c = 0.8$	0.894	2H0, $c = 0.8$	0.640	3H1, $c = 0.8$	0.184
(1H1, 1H2, 3), $c = 0.8$	1.069	(2H2, 2H3, 5), $c = 0.8$	0.599	(3H1, 3H2, 3), $c = 0.8$	0.164
(1H1, 1H2, 3), $c = 1.5$	0.931	(2H2, 2H3, 5), $c = 2.5$	0.104	(3H1, 3H2, 3), $c = 1.5$	0.146
(1H1, 1H2, 3), $c = 2.8$	0.360	(2H2, 2H3, 5), $c = 4.8$	0.199	(3H1, 3H2, 3), $c = 2.8$	0.093

which is stable at $[x_{i-3}, x_{i-1}]$ ($1 \leq c \leq 3$). The conservative forms (14) and (15) are applicable if

$$\begin{aligned}
 (\tilde{f}_E)_i^{3H2} &= \frac{1}{16}(-5\bar{\phi}_{i+1}^n + 11\bar{\phi}_i^n - 4\bar{\phi}_{i-1}^n - 4\bar{\phi}_{i-2}^n + \bar{\phi}_{i-3}^n + \bar{\phi}_{i-4}^n) \\
 (F_E)_i^{3H2} &= \frac{1}{48}c[(23 - 9c + c^2)\bar{\phi}_{i+1}^n - 3(7 - 7c + c^2)\bar{\phi}_{i-1}^n + (1 + 3c - c^2)\bar{\phi}_{i-3}^n]
 \end{aligned} \tag{38}$$

Summarizing stability regions of the derived schemes, it is clear that first-, third-, and higher-odd-order algorithms are stable within a single computational cell (two grid cells), i.e. $0 \leq \alpha \leq 1$ ($0 \leq c \leq 2$); while second-, fourth-, and higher-even-order ones are stable at two adjacent computational cells (four grid cells), i.e. $0 \leq \alpha \leq 2$ ($0 \leq c \leq 4$). Similar trend has been observed earlier by Leonard (2002) for the case of conventional methods. The derived schemes are compared using the square wave propagation test (Figure 7), and the respective error measure is shown in Table I. Application of the conventional methods such as 1UP, second-order central Lax–Wendroff (LW) [8, 9], and third-order upwind QUICKEST by Leonard [10] are shown for a comparison.

Computations confirmed that even-order schemes result in more dispersive algorithms having spurious trailing oscillations, whereas odd-order methods have better phase properties. Solution splitting of HOP methods is easily recognized by a saw-like behaviour due to usage of alternative node sets (odd and even). Increased truncation error, as predicted by Equation (13), is another drawback of the algorithms.

4. CONVEX COMBINATIONS OF HOPPING POLYNOMIALS

Recalling effectiveness of convex combinations of HOPs explored in earlier sections, it is expected that the linear operator

$$(\bar{\phi}_i^{n+1})^{(A,B,k)} = (1 - c/k)(\bar{\phi}_i^{n+1})^A + (c/k)(\bar{\phi}_i^{n+1})^B \tag{39}$$

is a good candidate to satisfy simultaneously several requirements: to increase computational time step, to ‘stitch’ the splitting solutions, to decrease the wiggling, and to increase overall accuracy. Here, superscripts A and B identify utilized approximations, and k is the parameter. Several combinations of A , B , and k , denoted as (A, B, k) , have been tested to satisfy the above requirements. Updates given by relationship (14b) are the best candidates for scheme A because expression (39) would have an exact solution $\bar{\phi}_i^n$ at $c = 0$; otherwise, if updates (14a) are used as A , the combination tends to the approximate value $\tilde{\phi}_i^n$ when $c \rightarrow 0$. Consequently, to utilize an alternative node set in combination (39), the B scheme must satisfy relationship (14a). Coefficient k is chosen to provide a smooth, accurate, and stable transition from scheme A to B when c increases from zero to its maximal stable value. Out of chosen combinations, the most promising updates (1H1, 1H2, 3), (2H2, 2H3, 5) and (3H1, 3H2, 3) are compared in Figure 7 and Table I.

First-order upwind update

$$(\bar{\phi}_i^{n+1})^{(1H1,1H2,3)} = (1 - c/3)(\bar{\phi}_i^{n+1})^{1H1} + (c/3)(\bar{\phi}_i^{n+1})^{1H2} \tag{40}$$

which can also be rewritten in conservative form (15) using east-face flux value

$$(f_E)_i^{(1H1,1H2,3)} = \frac{1}{6}c[(5 - c)\bar{\phi}_i^n + 2\bar{\phi}_{i-1}^n - (1 - c)\bar{\phi}_{i-2}^n] \tag{41}$$

is stable at $0 \leq c \leq 3$, as in Figure 8(a). Comparing with LxF scheme (1H0), the update (1H1, 1H2, 3) is less accurate at small Courant number, but more accurate for larger values of c . The scheme does not exhibit any solution splitting.

Second-order upwind scheme is given by the combination

$$(\bar{\phi}_i^{n+1})^{(2H2,2H3,5)} = (1 - c/5)(\bar{\phi}_i^{n+1})^{2H2} + (c/5)(\bar{\phi}_i^{n+1})^{2H3} \tag{42}$$

or in conservative form (15) by the east-face flux

$$(f_E)_i^{(2H2,2H3,5)} = \frac{1}{40}c[(c^2 - 11c + 38)\bar{\phi}_i^n - (3c - 23)\bar{\phi}_{i-1}^n - (2c^2 - 15c + 17)\bar{\phi}_{i-2}^n + (3c - 7)\bar{\phi}_{i-3}^n + (c^2 - 4c + 3)\bar{\phi}_{i-4}^n] \tag{43}$$

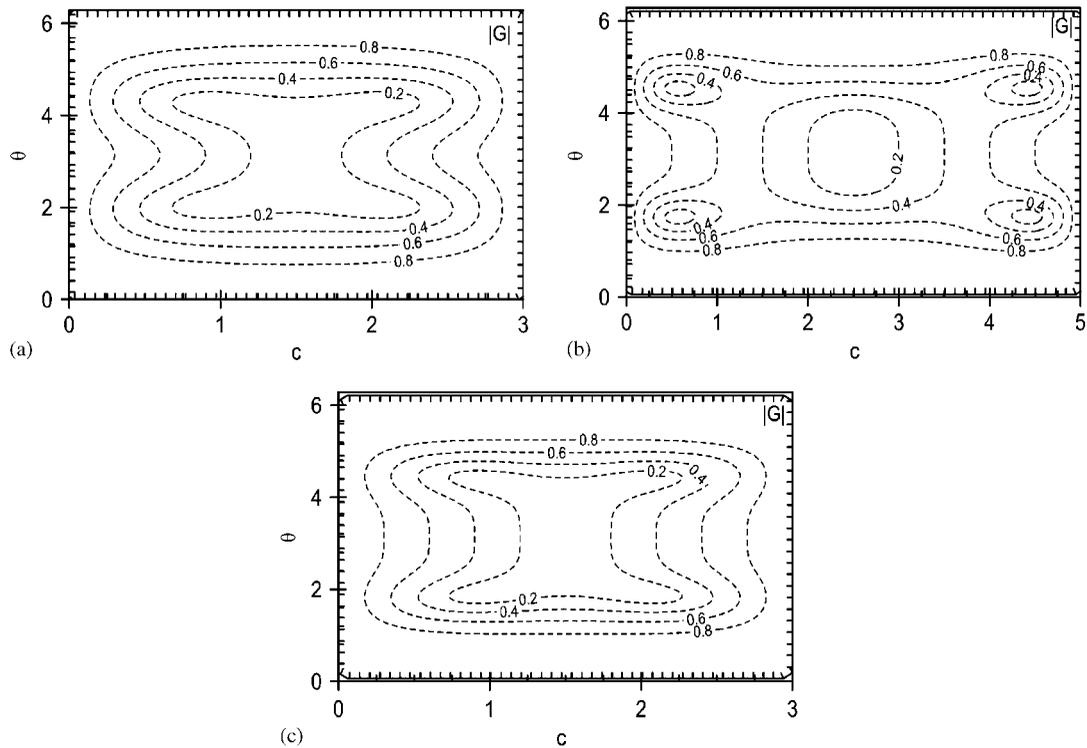


Figure 8. Amplification factor modulus $|G|$ for convex combinations of HOP schemes: (a) (1H1, 1H2, 3); (b) (2H2, 2H3, 5); and (c) (3H1, 3H2, 3).

The scheme is stable at $0 \leq c \leq 5$ (Figure 8(b)); it exhibits third-order properties for values c near centre of the interval, and is second-order otherwise.

Third-order scheme

$$(\bar{\phi}_i^{n+1})^{(3H1,3H2,3)} = (1 - c/3)(\bar{\phi}_i^{n+1})^{3H1} + (c/3)(\bar{\phi}_i^{n+1})^{3H2} \tag{44}$$

can be expressed in conservative form (15) using east-face flux

$$\begin{aligned} (f_E)_i^{(3H1,3H2,3)} = & \frac{1}{144}c[-(c^3 - 9c^2 + 26c - 24)\bar{\phi}_{i+2}^n - 3(c - 3)\bar{\phi}_{i+1}^n \\ & + 3(c^3 - 7c^2 + 7c + 31)\bar{\phi}_i^n + 48\bar{\phi}_{i-1}^n - 3(c^3 - 5c^2 - 2c + 8)\bar{\phi}_{i-2}^n \\ & + 3(c - 3)\bar{\phi}_{i-3}^n + (c^3 - 3c^2 - c + 3)\bar{\phi}_{i-4}^n] \end{aligned} \tag{45}$$

It is stable at $0 \leq c \leq 3$, as shown in Figure 8(c).

Generally, all derived combinations of HOP schemes show a greater truncation error for a small computational time step, as it is predicted by estimation (13); and the error reduces as the Courant number increases. For large values c , schemes (1H1, 1H2, 3) and (2H2, 2H3, 5) are more accurate than the conventional first- and second-order schemes; and, combination (3H1, 3H2, 3) is almost as accurate as the third-order QUICKEST. None of considered convex combinations exhibits solution splitting.

5. TWO-DIMENSIONAL HOPPING APPROXIMATIONS

The HOP solution technique is extendable for the 2-D hyperbolic equation

$$\frac{\partial \phi}{\partial t} + \frac{\partial u \phi}{\partial x} + \frac{\partial v \phi}{\partial y} = 0 \tag{46}$$

where $\mathbf{U} = (u, v)$ is the uniform velocity field in 2-D space (x, y) . Similar to the 1-D case (14), one searches for a solution of Equation (46) in the conservative finite-volume form

$$\begin{aligned} \bar{\phi}_{i,j}^{n+1} = & \bar{\phi}_{i,j}^n - (F_E)_{i,j} + (F_W)_{i,j} - (F_N)_{i,j} + (F_S)_{i,j} \\ (F_W)_{i,j} = & (F_E)_{i-2,j}, \quad (F_S)_{i,j} = (F_N)_{i,j-2} \end{aligned} \tag{47}$$

where bars indicate spatial overages over *computational cell* (i, j) , and the capital letter F still identifies DGC east-, west-, north-, and south-face fluxes $(F_E)_{i,j}$, $(F_W)_{i,j}$, $(F_N)_{i,j}$, and $(F_S)_{i,j}$.

For a computational efficiency, similar to the 1-D SGC formulation (15), expression (47) has to be rewritten in terms of SGC pseudo-fluxes as

$$\begin{aligned} \bar{\phi}_{i,j}^{n+1} = & \bar{\phi}_{i,j}^n - (f_E)_{i,j} + (f_W)_{i,j} - (f_N)_{i,j} + (f_S)_{i,j} \\ (f_W)_{i,j} = & (f_E)_{i-1,j}, \quad (f_S)_{i,j} = (f_N)_{i,j-1} \end{aligned} \tag{48}$$

Here, $(f_E)_{i,j}$, $(f_W)_{i,j}$, $(f_N)_{i,j}$, and $(f_S)_{i,j}$ are the east-, west-, north-, and south-face fluxes of the (i, j) th *grid cell* at time-level n , as in Figure 9(a).

To demonstrate a potential of extension of the HOP method for two dimensions, only one first-order scheme is derived; however, the technique could be followed for higher-order schemes

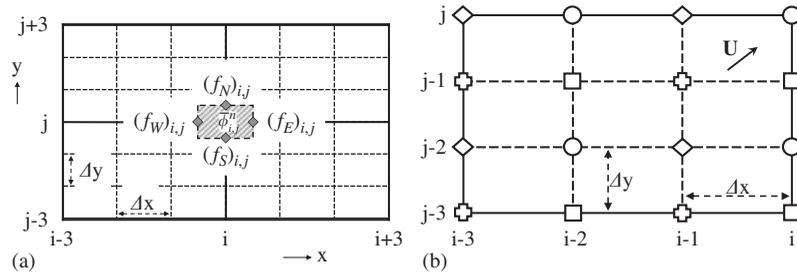


Figure 9. Computational stencil of 2-D HOP method for the velocity as shown. Each of the involved hopping polynomials utilizes four nodes identified as follows: $\circ - P_{i,j}$; $\square - P_{i,j-1}$; $\diamond - P_{i-1,j}$; and $\boxplus - P_{i-1,j-1}$.

as well. Analogically to the procedure established earlier in the paper, the following HOPs are introduced as (for positive \mathbf{U} , as in Figure 9(b))

$$\begin{aligned}
 P_{i,j} &= \frac{1}{4}[(2-c)(2-d)\bar{\phi}_{i,j}^n + c(2-d)\bar{\phi}_{i-2,j}^n \\
 &\quad + (2-c)d\bar{\phi}_{i,j-2}^n + cd\bar{\phi}_{i-2,j-2}^n] \\
 P_{i-1,j} &= \frac{1}{4}[(3-c)(2-d)\bar{\phi}_{i-1,j}^n - (1-c)(2-d)\bar{\phi}_{i-3,j}^n \\
 &\quad + (3-c)d\bar{\phi}_{i-1,j-2}^n - (1-c)d\bar{\phi}_{i-3,j-2}^n] \\
 P_{i,j-1} &= \frac{1}{4}[(2-c)(3-d)\bar{\phi}_{i,j-1}^n + c(3-d)\bar{\phi}_{i-2,j-1}^n \\
 &\quad - (2-c)(1-d)\bar{\phi}_{i,j-3}^n - c(1-d)\bar{\phi}_{i-2,j-3}^n] \\
 P_{i-1,j-1} &= \frac{1}{4}[(3-c)(3-d)\bar{\phi}_{i-1,j-1}^n - (1-c)(3-d)\bar{\phi}_{i-3,j-1}^n \\
 &\quad - (3-c)(1-d)\bar{\phi}_{i-1,j-3}^n + (1-c)(1-d)\bar{\phi}_{i-3,j-3}^n]
 \end{aligned}
 \tag{49}$$

Here, $c = u\Delta t/\Delta y$ and $d = v\Delta t/\Delta y$ are the components of 2-D Courant number $\mathbf{C} = (c, d)$. Each of polynomials (49) represents bilinear interpolant hopping over one row and one column of nodes as shown in Figure 9(b). More compact form of expressions (49) reads

$$\begin{aligned}
 P_{i,j} &= E_{i,j} \\
 P_{i-1,j} &= E_{i-1,j} + R_{i-1,j} + dD_{i-1,j} \\
 P_{i,j-1} &= E_{i,j-1} + S_{i,j-1} + cD_{i,j-1} \\
 P_{i-1,j-1} &= E_{i-1,j-1} + R_{i-1,j-1} + S_{i-1,j-1} + (c+d-1-cd)D_{i-1,j-1}
 \end{aligned}
 \tag{50}$$

where

$$\begin{aligned}
 E_{i,j} &= \bar{\phi}_{i,j}^n - cR_{i,j} - dS_{i,j} - cdD_{i,j} \\
 R_{i,j} &= (\bar{\phi}_{i,j}^n - \bar{\phi}_{i-2,j}^n)/2, \quad S_{i,j} = (\bar{\phi}_{i,j}^n - \bar{\phi}_{i,j-2}^n)/2 \\
 D_{i,j} &= -(\bar{\phi}_{i,j}^n - \bar{\phi}_{i-2,j}^n - \bar{\phi}_{i,j-2}^n + \bar{\phi}_{i-2,j-2}^n)/4
 \end{aligned}$$

Derived relationships can be extended straightforwardly to arbitrary velocities by changing signs ‘-’ to ‘+’ in subscript space indices in the case of negative \mathbf{U} .

According to the von Neumann stability analysis, the single-step updates

$$\begin{aligned}(\bar{\phi}_{i,j}^{n+1})^{1H1,1} &= P_{i,j} \\(\bar{\phi}_{i,j}^{n+1})^{1H2,1} &= P_{i-1,j} \\(\bar{\phi}_{i,j}^{n+1})^{1H1,2} &= P_{i,j-1} \\(\bar{\phi}_{i,j}^{n+1})^{1H2,2} &= P_{i-1,j-1}\end{aligned}\tag{51}$$

are stable within rectangles identified by the lower left and upper right corners as $[(i-2, j-2), (i, j)]$, $[(i-2, j-3), (i, j-1)]$, $[(i-3, j-2), (i-1, j)]$, and $[(i-3, j-3), (i-1, j-1)]$, respectively (Figure 9(b)). It is obvious that the scheme ‘1H1, 1’ reduces to ‘1H1’ for a 1-D case; and ‘1H2, 2’ becomes ‘1H2’. Similar to the 1-D cases, 2-D schemes (51) exhibit saw-like solutions due to usage of alternative node sets (odd and even); therefore, a convex combination of the schemes is expected to ‘stitch’ the splitting solutions, increase maximal stable Courant number and to improve overall accuracy of the resulting algorithm. Following the 1-D experience (Equation (39)), a bilinear interpolant

$$\bar{\phi}_{i,j}^{n+1} = \frac{1}{9}[(3-c)(3-d)P_{i,j} + c(3-d)P_{i-1,j} + (3-c)dP_{i,j-1} + cdP_{i-1,j-1}]\tag{52}$$

is expected to fulfil the requirements. Update (52) is the sought 2-D first-order upwind HOP scheme. To satisfy conservative form (48), terms in Equation (52) must be regrouped in order to recover expressions for the fluxes. The fluxes are not ‘true’ and unique in the spirit of the flux integral method by Leonard *et al.* [11], because the flux splitting procedure does not guarantee unique flux expressions in the 2-D case. However, it does not mean that a successful particular solution cannot be found. One solution is obtained as

$$\begin{aligned}(f_E)_{i,j} &= \frac{1}{18}c[((6-d)E_{i,j} + dE_{i,j-1}) + 2d(S_{i,j-1} - S_{i,j} - S_{i-1,j}) \\&\quad + d(5D_{i,j} - (1-2c)D_{i,j-1}) + 6(R_{i,j} + \bar{\phi}_{i,j}^n + \bar{\phi}_{i-1,j}^n)] \\(f_N)_{i,j} &= \frac{1}{18}d[((6-c)E_{i,j} + cE_{i-1,j}) + 2c(R_{i-1,j} - R_{i,j} - R_{i,j-1}) \\&\quad + c(5D_{i,j} - (1-2d)D_{i-1,j}) + 6(S_{i,j} + \bar{\phi}_{i,j}^n + \bar{\phi}_{i,j-1}^n)]\end{aligned}\tag{53}$$

To conserve mass, western and southern fluxes are taken as $(f_W)_{i,j} = (f_E)_{i-1,j}$ and $(f_S)_{i,j} = (f_N)_{i,j-1}$, respectively.

The derived 2-D HOP method (Equations (48) and (53)) is thoroughly tested. One of the tests is a complete round of a clockwise rotation of the ‘Gaussian hill’ (Figure 10 and Table II). If solved exactly, the ‘hill’ returns to its original place maintaining the initial form. A numerical approximation introduces relative error

$$\varepsilon = \frac{\sum_{i,j} |\bar{\phi}_{i,j} - \phi_{i,j}^*|}{\sum_{i,j} |\phi_{i,j}^*|}\tag{54}$$

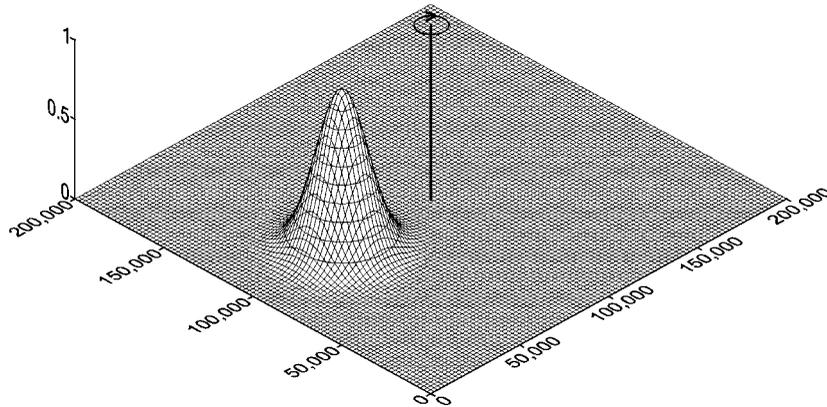


Figure 10. Initial profile for 2-D test case.

Table II. Performance of 2-D first-order upwind and HOP scheme using the rotational flow test with different maximal Courant number.

$\Delta x, \Delta y$ (km)	1UP, $C_{\max} = 1$			HOP, $C_{\max} = 1$			HOP, $C_{\max} = 3$		
	$\Delta\phi$	ε	δt	$\Delta\phi$	ε	δt	$\Delta\phi$	ε	δt
2	0.675	0.776	1	0.751	0.928	1.002	0.677	0.779	0.348
1	0.513	0.516	1	0.602	0.657	1.019	0.513	0.516	0.347
0.5	0.347	0.311	1	0.433	0.416	1.094	0.347	0.311	0.370
0.25	0.211	0.174	1	0.278	0.242	1.257	0.211	0.174	0.419

Note: Maximum absolute error ($\Delta\phi$), error measure (ε), and relative required computational time (δt) are compared.

where ϕ^* is the exact solution, and $\bar{\phi}$ is the numerical one. Maximal absolute error

$$\Delta\phi = \max_{i,j} |\bar{\phi}_{i,j} - \phi_{i,j}^*| \tag{55}$$

is also an important characteristic. To perform a grid refinement test, the constant-depth domain (200 × 200) km is covered by a mesh with the grid sizes $\Delta x = \Delta y = (2, 1, 0.5$ and $0.25)$ km, leading to computational grids with (101 × 101), (201 × 201), (401 × 401) and (801 × 801) nodes, respectively. Two different time steps are considered, to provide the maximal Courant number of 1 and 3. Execution time and solution of the 2-D HOP method is compared with the conventional 2-D 1UP method.

Numerical experiments show that the 2-D HOP method is stable within a large square $0 \leq c \leq 3$, $0 \leq d \leq 3$, comparing with the conventional stability criterion $0 \leq c \leq 1$, $0 \leq d \leq 1$ of the conventional 1UP scheme. Even though, the execution time of the HOP method is slightly longer for the Courant number near or below a unity, the algorithm is up to three times faster than 1UP for large C . HOP is slightly more diffusive for small time steps, and is equally accurate with 1UP at large C (Figure 11). Overall, the HOP method demonstrates a good performance and high efficiency.

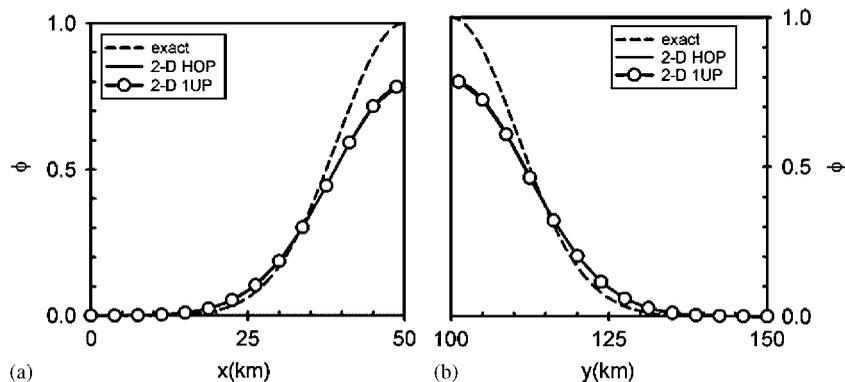


Figure 11. Exact and computed solutions of the 2-D Gaussian hill rotation using first-order upwind and HOP schemes: (a) x cross-section and (b) y cross-section.

6. CONCLUSIONS

Using hopping polynomial (HOP) approximations of up to third-order of accuracy, a set of new explicit numerical methods has been developed for the hyperbolic equation solution. To widen stability interval, HOP stencils skip odd or even nodes of a computational grid. Derived odd- and even-node-based updates are convexly combined to overcome the solution splitting, and to improve overall accuracy and stability of the resulting scheme. Derived explicit odd-order approximations are stable for the Courant number ranging from 0 to 3, and even-order schemes are stable at for up to $C = 5$. A straightforward technique of HOP method extension for two dimensions is suggested using first-order upwind algorithm as an example. Computational tests show a substantial robustness, efficiency and accuracy of the HOP method.

REFERENCES

1. Harten A, Osher S. Uniformly high-order accurate nonoscillatory schemes. I. *SIAM Journal on Numerical Analysis* 1987; **24**(2):279–309.
2. Liu X-D, Osher S, Chan T. Weighted essentially non-oscillatory schemes. *Journal of Computational Physics* 1994; **115**(1):200–212.
3. Lax PD. Weak solutions of non-linear hyperbolic equations and their numerical computations. *Communications on Pure and Applied Mathematics* 1954; **7**:159–193.
4. Leonard BP. Stability of explicit advection schemes. The balance point location rule. *International Journal for Numerical Methods in Fluids* 2002; **38**:471–514.
5. Richards FB. A Gibbs phenomenon for spline functions. *Journal of Approximation Theory* 1991; **66**:334.
6. Fejér L. Untersuchungen über fouriersche reihen. *Mathematische Annalen* 1903; **58**:51–69.
7. Lanczos C. *Discourse on Fourier Series*. Oliver & Boyd: Edinburgh, 1966.
8. Lax PD, Wendroff R. Systems of conservation laws. *Communications on Pure and Applied Mathematics* 1960; **13**:217–237.
9. Lax PD, Wendroff R. Difference schemes with high order accuracy for solving hyperbolic equations. *Communications on Pure and Applied Mathematics* 1964; **17**:381.
10. Leonard BP. A stable and accurate convective modeling procedure based on quadratic upstream interpolation. *Computer Methods in Applied Mechanics and Engineering* 1979; **19**:59–98.
11. Leonard BP, MacVean MK, Lock AP. The flux integral method for multidimensional convection and diffusion. *Applied Mathematical Modelling* 1995; **19**:333–342.