An unstructured finite volume method for large-scale shallow flows using the fourth-order Adams scheme

Abdelaziz Beljadid a, Abdolmajid Mohammadian a, Hazim M. Qiblawey b

a Department of Civil Engineering, University of Ottawa, 161 Louis Pasteur, Ottawa, Ontario K1N6N5, Canada
b Department of Chemical Engineering, Qatar University, P.O. Box 2713, Doha, Qatar

A R T I C L E   I N F O

Article history:
Received 14 October 2012
Received in revised form 13 August 2013
Accepted 15 October 2013
Available online 24 October 2013

Keywords:
Shallow water
Rossby waves
Finite volume method
Coriolis effect
Adams method
Operator splitting

A B S T R A C T

In this paper, we introduce a new upwind finite volume method using unstructured grids for large-scale shallow flows. This method uses a high-order upwind scheme for the calculation of the numerical flux, and the fourth-order Adams method with a splitting approach for time integration. The process includes three stages: in the first and third steps the Coriolis term is integrated analytically, and in the second step the flux term is integrated numerically. Most upwind schemes perform well for gravity waves but they lead to a high level of damping or numerical oscillations for Rossby waves. The proposed method presents the advantage that it performs well for both gravity and Rossby waves. The use of fourth-order Adams method without any iteration on the corrector is enough to suppress the short-wave numerical noise without damping the long waves that are essential in the transport of energy Rossby waves, in large-scale oceanic and atmospheric flows.

1. Introduction

Shallow water equations (SWEs) are used to describe many physical phenomena in oceans, rivers, the atmosphere, etc. These equations are applicable when the vertical velocity component is negligible compared to the horizontal components, and are obtained by assuming hydrostatic pressure distribution (e.g., [29]). The three-dimensional incompressible Navier–Stokes equations are averaged over the depth to obtain the SWEs. In the absence of viscous terms, SWEs can be considered a hyperbolic system. The finite volume (FV) methods are most convenient for modeling these systems since they have a conservative form. Upwind finite volume (UFV) methods can numerically solve these systems with good accuracy and an acceptable computational cost.

UFV schemes use exact or approximate methods to solve the Riemann problem at the interface of computational cells. Godunov’s method [11, 10] is the most popular scheme using the exact solution of the Riemann problem. Its extension to second-order and to high-order schemes is given by Van Leer [27] and Colella and Woodward [7], respectively. The exact algorithms are computationally expensive compared to the approximate methods. Roe’s method [22], which is applied in this work, is the most popular approximate method. It requires an accurate estimation of parameter values near the interface on both sides of the computational cell. In the presence of source terms in the SWEs, the UFV schemes may lead to numerical oscillations due to the imbalance between the source and flux terms. To overcome this problem, some special treatments can be applied for balancing the source and flux terms. A large number of studies have been conducted in this direction, such as Vázquez-Cendón [28], Gallouet et al. [9], Mohammadian et al. [21], Mohammadian and Le Roux [20], and Stewart et al. [25]. Other studies have been conducted to evaluate the performance of various schemes for large-scale shallow flows (e.g., [31, 8, 14, 13, 17, 12, 30, 16]). Nevertheless, UFV methods are considered in a limited number of studies (e.g., [18, 19, 6, 1]). The performance of numerical methods is greatly influenced by the temporal schemes used. Total Variation Diminution (TVD) temporal integration methods, developed by Shu and Osher [24], are among the most popular temporal integration schemes. They are widely used for their ability to avoid oscillations and to maintain stability. Furthermore, some higher-order TVD schemes are insensitive to the values of Courant–Friedrichs–Lewy (CFL) numbers and present highly accurate results over a wide range of CFL numbers [2].

Beljadid et al. [2] studied the performance of UFV schemes with TVD Runge–Kutta methods for temporal integration. Several aspects were examined, including mass and energy conservation, numerical diffusion, and numerical oscillations for Kelvin, Yanai, Poincaré, and gravity waves. The accuracy of various schemes was analyzed for different types of waves in order to identify the most accurate and efficient numerical schemes. Through numerical
experiments, it was demonstrated that a third-order TVD Runge-Kutta method (TVDRK3) combined with the upwind-centered scheme provides accurate results for Kelvin, Yanai, Poincaré, gravity, and inertia gravity waves. The TVDRK3 method with the upwind-centered scheme was found to be a good choice for these types of waves. It was shown that the results remained accurate for a wide range of CFL numbers, which is important in practical applications. Moreover, this scheme presents good stability properties even for large spatial variation of computational cells, usually present in unstructured grids. However, this method fails in the modeling of Rossby waves, which have a particular behavior and are difficult to capture by several well-known upwind schemes. In this paper we propose a new upwind finite volume method which presents a good improvement for the modeling of Rossby waves. A high-order spatial scheme based on polynomial fitting is proposed. Operator splitting and the fourth-order Adams method are used for temporal integration.

The paper is organized as follows: SWEs are presented in Section 2. In Section 3, the proposed finite volume method is described. Section 4 presents some numerical experiments for equatorial Rossby waves. In Section 5, some numerical experiments are performed using the proposed method for nonlinear SWEs. Some concluding remarks complete the study.

2. Shallow water equations

In this section, linear and nonlinear shallow water equations are presented. The conservative form of the 2D shallow water equations is written as [29]:

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{E}}{\partial x} + \frac{\partial \mathbf{G}}{\partial y} = \mathbf{S}$$

(1)

The linear and nonlinear equations are defined in terms of parameters \( \mathbf{U}, \mathbf{E}, \mathbf{G} \) and \( \mathbf{S} \).

2.1. Linear SWEs

For linear shallow water equations, the parameters \( \mathbf{U}, \mathbf{E}, \mathbf{G} \) and \( \mathbf{S} \) are defined as:

$$\mathbf{U} = \begin{bmatrix} \eta \\ u \\ v \end{bmatrix}, \quad \mathbf{E} = \begin{bmatrix} Hu \\ g\eta_h \\ 0 \end{bmatrix}, \quad \mathbf{G} = \begin{bmatrix} Hv \\ 0 \\ g\eta_v \end{bmatrix}, \quad \mathbf{S} = \begin{bmatrix} 0 \\ f \nu \\ -fu \end{bmatrix}.$$  

(2)

where \( \eta \) represents the water surface elevation, \( u \) and \( v \) are the depth-averaged velocity components in the \( x \)- and \( y \)-directions, respectively, \( f \) is the Coriolis parameter, \( g \) is the gravity acceleration, and \( (H + \eta) \) is the total water depth.

The term \( \mathbf{S} \) may include various source terms such as bed friction, bed topography, and wind stress. Since this paper concentrates on Rossby waves, the source term \( \mathbf{S} \) is assumed to include the Coriolis parameter.

The beta-plane approximation to the Coriolis parameter is considered \( f = \frac{\beta}{\lambda} \), where \( \beta \) is the linear coefficient of variation of \( f \) with respect to \( y \). The variable \( y \) is considered as the meridional distance from the equator (positive northward). The parameter \( \beta \) is given as:

$$\beta = 2\Omega R = 2.29 \times 10^{-11} \text{m}^{-1}\text{s}^{-1}$$

(3)

where \( \Omega \) and \( R \) are the angular speed of the Earth’s rotation and the mean radius of the Earth, respectively \( (\Omega = 7.29 \times 10^{-5} \text{rad s}^{-1}, R = 6371 \text{km}) \).

The dimensionless form of SWEs is used in this paper. The model Eqs. (1) and (2) are converted into dimensionless form on an equatorial beta-plane using the variables \( \hat{x} = x/L', \quad \hat{y} = y/L', \quad \hat{h} = h/H', \quad \hat{u} = u/U', \quad \hat{v} = v/V' \). The reference values of the depth \( (H') \), time \( (T') \), length \( (L') \) and velocity \( (U') \) scales are expressed as:

$$H' = H, \quad T' = \frac{1}{\beta T}, \quad L' = \frac{1}{\beta L}, \quad U' = \frac{V'}{T'}$$

(4)

The resulting system, the Jacobian matrix, and the corresponding eigenvalues and eigenvectors are given in Appendix A.

2.2. Nonlinear SWEs

For nonlinear shallow water equations, the parameters \( \mathbf{U}, \mathbf{E}, \mathbf{G} \) and \( \mathbf{S} \) are defined as:

$$\mathbf{U} = \begin{bmatrix} h \\ hu \\ hv \end{bmatrix}, \quad \mathbf{E} = \begin{bmatrix} hu + 0.5gh^2 \\ hv \\ hv \end{bmatrix}, \quad \mathbf{G} = \begin{bmatrix} hv \\ hu \\ h^2 + 0.5gh^2 \end{bmatrix}$$

(5)

The source term \( \mathbf{S} \) is assumed to include the Coriolis effect

$$\mathbf{S} = (0, fhv, -fhv)^t$$

(6)

where \( h \) is the total fluid depth.

In the presence of the Coriolis effect, the nonlinear SWEs are converted into a dimensionless form on an equatorial beta-plane using the variables \( \hat{x} = x/L', \quad \hat{y} = y/L', \quad \hat{h} = h/H', \quad \hat{u} = u/U', \quad \hat{v} = v/V' \). The characteristic time \( (T') \), length \( (L') \) and velocity \( (U') \) scales are expressed in terms of the parameter \( \beta \) in the same way using Eq. (4), where the parameter \( H' \) is the mean water depth.

When the Coriolis force is absent, the following reference parameters are used to convert the nonlinear SWEs to a dimensionless form:

$$T' = L' \sqrt{\frac{gh}{\beta}}; \quad U' = V' \frac{L'}{T'}$$

(7)

where the characteristic length \( L' \) can be arbitrarily chosen and the parameter \( H' \) can be chosen with the same order as the mean water depth \( h \) in the system.

3. Finite volume method

An upwind finite volume method on an unstructured grid is employed in this paper. The variables are located at the geometric centers of the computational grids. Each triangle represents a control volume. The SWEs are integrated over every control volume as:

$$\int_\Gamma \left( \frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{E}}{\partial x} + \frac{\partial \mathbf{G}}{\partial y} \cdot \mathbf{S} \right) d\Omega = 0$$

(8)

where \( \Gamma \) and \( \Omega \) denote the boundary and the area of the domain, respectively.

By using the divergence theorem, the flux integral is transformed into a boundary integral:

$$\int_\Gamma \left( \frac{\partial \mathbf{E}}{\partial x} + \frac{\partial \mathbf{G}}{\partial y} \right) d\Gamma = \int_{\mathbf{F}} \mathbf{F} \cdot \mathbf{n} d\Gamma$$

(9)

where \( \mathbf{F} = (\mathbf{E}, \mathbf{G})^t \) is the flux vector and \( \mathbf{n} \) is the unit outward normal vector to the boundary \( \Gamma \). Then, (8) leads to
\[
\frac{d}{dt} \int_{\Omega} \mathbf{U} d\Omega + \int_{\Gamma} \mathbf{F} \cdot \mathbf{n} \, d\Gamma = \int_{\Gamma} \mathbf{S} d\Omega \tag{10}
\]

3.1. Unstructured grid implementation

For an unstructured triangular grid, the boundary integral \( \int_{\Gamma} \mathbf{F} \cdot \mathbf{n} \, d\Gamma \) in (10) may be approximated by a summation over the triangle edges as:

\[
\int_{\Gamma} \mathbf{F} \cdot \mathbf{n} \, d\Gamma = \sum_{k=1}^{3} \int_{\Gamma_k} \mathbf{F} \cdot \mathbf{n} \, d\Gamma_k = \sum_{k=1}^{3} (\mathbf{F}_{i_k} \cdot \mathbf{n}_k) \, l_k, \tag{11}
\]

where \( \Gamma_k, \mathbf{F}_k, \mathbf{n}_k \), and \( l_k, \) \( k = 1, 2, 3 \), are respectively the triangle edges, the outward fluxes, the unit outward normal vectors, and the lengths corresponding to the edges of a triangular cell.

The convective flux \( \mathbf{F} \) can be calculated by various schemes. Most schemes may be written in a general form as:

\[
\mathbf{F} = 0.5(\mathbf{F}_L + \mathbf{F}_R - \Delta \mathbf{F}^s), \tag{12}
\]

where \( \mathbf{F}_L = \mathbf{F}(U_L) \) and \( \mathbf{F}_R = \mathbf{F}(U_R) \) are the left and right flux vectors.

The flux difference \( \Delta \mathbf{F}^s \), which plays the role of stabilization, is computed based on Roe's linearization:

\[
\Delta \mathbf{F}^s = \sum_{k=1}^{3} \tilde{a}_k \tilde{e}_k \mathbf{e}_k \tag{13}
\]

where \( \tilde{a}_k, \mathbf{e}_k, \) and \( \tilde{e}_k \) are, respectively, the eigenvalues and the eigenvectors of the approximate Jacobian \( J \) and the coefficients of decomposition of \( \Delta \mathbf{U} = \mathbf{U}_e - \mathbf{U}_i \) on the basis of the eigenvectors, as explained in Appendix A.

In the \( \kappa \) scheme, \( \mathbf{U}_L \) and \( \mathbf{U}_R \) are calculated at the interface as:

\[
\begin{align*}
\mathbf{U}_L &= \mathbf{U}_w + \frac{\delta}{2} [(1 - K \kappa) (\mathbf{U}_w - \mathbf{U}_{aw}) + \kappa (1 + K \kappa) (\mathbf{U}_e - \mathbf{U}_w)] \\
\mathbf{U}_R &= \mathbf{U}_e + \frac{\delta}{4} [(1 - K \kappa) (\mathbf{U}_e - \mathbf{U}_{aw}) + \kappa (1 + K \kappa) (\mathbf{U}_w - \mathbf{U}_e)],
\end{align*}
\tag{14}
\]

with \( \delta = \frac{2 \Delta}{
\left| \mathbf{L}_{aw} \right| \left| \mathbf{L}_{e} \right|}, \) where \( \mathbf{L}_{aw} \) and \( \mathbf{L}_{e} \) are defined in Fig. 1. The slope limiter \( s \) is calculated in this paper using:

\[
s = \frac{2 \Delta_+ A_+}{A_+^2 + A_+^2 + \varepsilon}, \quad \varepsilon > 0 \tag{15}
\]

with \( A_+ = \mathbf{U}_e - \mathbf{U}_{aw} \) and \( A_- = \mathbf{U}_w - \mathbf{U}_e \).

The parameter \( \varepsilon \) is a small positive number chosen according to the order of the scheme. It should be limited in order to not affect the order of accuracy of the numerical scheme. In the numerical experiments presented in this paper, for a numerical scheme of order \( r \), the parameter \( \varepsilon \) is chosen as \( 0 < \varepsilon < \Omega_{m}^{-1} \), where \( \Omega_{m} \) is the area of the smallest cell in the entire computational domain.

Depending on \( \kappa \), Eq. (14) lead to the following schemes:

0. simplified Fromm scheme,
1/6. cell-based third order upwind method,
1/3. third-order upwind method,
1/2. Quick scheme,
1. upwind-centered scheme

The case \( \kappa = -1 \) corresponds to the second-order upwind scheme. This scheme is not considered in this paper since it leads to inaccurate results for Kelvin, Yanai, and Poincaré waves [2].

3.2. The proposed high-order upwind scheme

The method introduced in this paper includes a high-order upwind interpolation scheme. Upwind methods can be improved if the values of the parameters on both sides of the interface are estimated with more accuracy. The proposed method uses polynomials with two variables. In this paper, we use the third-order Lagrange polynomials in \( \sigma \) and linear interpolation in \( \tau \), where the variable \( \sigma \) denotes the axis perpendicular to the interface and \( \tau \) coincides with the interface as they are shown in Fig. 2. The value of \( \mathbf{U}_I \) is obtained by interpolation using three grid points upstream of the interface and one grid point downstream of the interface. The parameter \( \mathbf{U}(\sigma, \tau) \) is obtained by the following interpolation:

\[
\mathbf{U}(\sigma, \tau) = \sum_{i=1}^{j-4} L_i^{(1)}(\sigma) (\mathbf{U}_i - \mathbf{Q}_i^{(1)}(\tau_i)) + \mathbf{Q}_i^{(1)}(\tau)
\tag{16}
\]

where \( \mathbf{Q}_i^{(1)}(\tau) \) is a polynomial which depends on the variable \( \tau \) and \( L_i^{(1)} \) is the Lagrange polynomial associated with the cell \( i \), obtained from the parameters of three cells on the left-hand side and one cell on the right-hand side of the interface.

\[
L_i^{(1)}(\sigma) = \prod_{j=1, j \neq i}^{j-4} \frac{(\sigma - \sigma_j)}{(\sigma_i - \sigma_j)}
\tag{17}
\]

The parameter \( \mathbf{U}_i \) is obtained by integration along the left hand-side of the interface as:
by using Lagrange polynomials, we obtain in the same way as for $\sigma^+$ and $\sigma^-$, can thus be obtained by solving the following system:

$$
\mathbf{U}(\sigma, \tau) = \sum_{i=1}^{N} L_i^0(\sigma) (\mathbf{U}_i - Q^0(\tau_i) + Q^0(\tau))
$$

The form of the above equation allows defining the polynomial $Q^0(\tau)$ without a constant. Then, for the linear case, the polynomial $Q^0(\tau)$ is obtained by integration along the right-hand side value $U_l$ of the interface and the slope $N_l$ of the computational cell. Since $N_l = 2$, the parameter $U_l$ is obtained by solving the following system:

$$
\left\{ \begin{array}{l}
\frac{\partial U}{\partial \tau} = 0 \\
\frac{\partial U}{\partial \rho} = 0 \\
\frac{\partial U}{\partial \beta} = 0
\end{array} \right.
$$

The parameters at the vertex $P_2$ are obtained in the same way as for $P_1$.

### 3.3. Temporal integration method

In most finite volume models, a Runge–Kutta method is used for temporal integration. A popular approach is the TVD Runge–Kutta method, which is explained in Appendix A. In Beljadid et al. [2], it was shown that $\kappa$ schemes combined with the TVD Runge–Kutta method lead to good results for Kelvin, Yanai, gravity, and inertia gravity waves, and that the upwind-centered scheme is the best one. The third-order TVD Runge–Kutta method combined with the $\kappa$ scheme leads to inaccurate results for Rossby waves due to an imbalance between the flux and Coriolis terms. In this paper, the fourth-order Adams method is proposed for the temporal scheme, with operator splitting for the Coriolis and flux terms. The process includes three stages: in the first and third steps the Coriolis term is integrated analytically, and in the second step the flux term is integrated numerically. In the following, the temporal integration method is explained for linear SWEs. The method for nonlinear SWEs can be achieved by replacing $u$ and $v$ with $hu$ and $hv$, respectively. Following Beljadid et al. [1], first, the effect of the source term (the Coriolis effect) is considered:

$$
\frac{\partial \mathbf{U}}{\partial t} = -\mathbf{S}
$$

Then, the other terms are added:

$$
\frac{\partial \mathbf{U}}{\partial t} = \mathbf{F}
$$

where

$$
\mathbf{F} = -\left( \frac{\partial \mathbf{E}}{\partial x} + \frac{\partial \mathbf{G}}{\partial y} \right)
$$

The system with Coriolis term only, i.e.,

$$
\frac{\partial u}{\partial t} = f \nu \\
\frac{\partial v}{\partial t} = -fu \\
\frac{\partial \eta}{\partial t} = 0
$$

can be solved analytically as:

$$
u = v_0 \cos(ft) + v_0 \sin(ft) \\
\nu' = v_0' \cos(ft) - v_0' \sin(ft) \\
\eta' = \eta^n
$$

In the following, $\Delta t$ represents the time-step size and $\eta^n, u^n, v^n$ are respectively the water surface elevation and the $x$- and $y$-velocities at time $t^n$. First, (28) is integrated over half of the time step:

$$
\begin{align}
\frac{u'}{t^n} &= u^n \cos(f \Delta t / 2) + v^n \sin(f \Delta t / 2) \\
\frac{v'}{t^n} &= v^n \cos(f \Delta t / 2) - u^n \sin(f \Delta t / 2) \\
\eta' &= \eta^n
\end{align}
$$

Then, (29) is integrated over the entire time step using the fourth-order Adams method, as explained later. The result after
The fourth-order Adams method (Appendix B) uses the fourth-order explicit Adams–Bashforth scheme as the predictor, and the fourth-order Adams–Moulton method for the corrector, as explained below. We denoted by \( \mathbf{f}' = \sum_{i=1}^{3} f_i \cdot \mathbf{n}_i \) \( i \) the result of the integration of the flux on the control volume at time \( t_n = n\Delta t \).

The result \( \mathbf{U}' \) of equations (33) is used to obtain the predictor value:

\[
\mathbf{U}^{\text{pred}} = \mathbf{U}' + \Delta t\left( \frac{5}{24} \mathbf{f}' + \frac{19}{24} \mathbf{f}^{\text{pred}} + \frac{5}{24} \mathbf{f}^{\text{pred-1}} + \frac{1}{24} \mathbf{f}^{\text{pred-2}} \right)
\]  

where \( \mathbf{f}' \) is the result of the integration of the flux using \( \mathbf{U}' \).

The corrector value is obtained as:

\[
\mathbf{U}^{\text{corr}} = \mathbf{U}' + \Delta t\left( \frac{9}{24} \mathbf{f}^{\text{pred}} + \frac{19}{24} \mathbf{f}' - \frac{5}{24} \mathbf{f}^{\text{pred-1}} + \frac{1}{24} \mathbf{f}^{\text{pred-2}} \right)
\]  

where \( \mathbf{f}^{\text{pred}} \) is the result of the integration of the flux using the predictor value \( \mathbf{U}^{\text{pred}} \).

In the numerical experiments presented below, we will show that it is not advantageous to consider any additional iteration or modifier at the corrector step, and that even with only one iteration, optimal results are obtained. Note that in the above algorithm, the integration of the flux term (the values \( \mathbf{f}' \) and \( \mathbf{f}^{\text{pred}} \)) which is the most computationally expensive part, is required only twice. Therefore, the above version of the fourth-order Adams method is less expensive than the fourth-order Runge–Kutta method (RK4), in which the flux must be integrated four times for each step. We used an Intel Core i7 2670QM in our calculations. For the fourth-order Adams–Bashforth predictor formula (35) and (36), the CPU time for the Adams method is 191 while the CPU time is 361 for the RK4 method.

Note that the proposed method solves exactly stationary solutions corresponding to water at rest (\( h = \text{Constant} \) and \( u = v = 0 \)) for systems (1) and (2). If this solution is considered as the initial condition, the operations of interpolation introduced in Section 3.2, for the spatial scheme of the proposed method, preserve the constant solutions and lead to flux equalizing zero. The splitting before and after treatment of the flux does not affect the solution, since the condition \( u = v = 0 \) gives the same solution for equations (33) and (34). Then the parameters \( k_1, k_2, k_3 \) and \( k_4 \) which are used in the fourth-order Adams–Bashforth predictor formula and the Adams–Moulton corrector formula are zero (Eqs. (35) and (36)). Similarly, \( k_i = 0 \), \( i = 1, 2, 3, 4 \), are the coefficients used in RK4 (Eqs. (60) and (61) in Appendix B2) which is used to start the method in the first three steps. Then the constant solutions are preserved, which is also confirmed by the numerical results (not shown). In the numerical tests presented below, we also examine the well-balanced property for non-trivial steady state solutions. In particular, the steady state solutions in a moving frame, such as the exact solutions of the linear beta-plane equatorial shallow water equations, are considered.

4. Numerical experiments for symmetric linear equatorial Rossby waves

In this section, we perform a test for linear beta-plane equatorial shallow water equations by using large-scale equatorial waves, and test the ability of the proposed method to capture the slow waves. In particular, we consider the symmetric equatorial Rossby waves of index 1, which are exact solutions of the linear beta-plane equatorial shallow water equations.

4.1. Symmetric equatorial Rossby waves of index 1

Equatorial Rossby waves, also called planetary waves, are found near the equator. They propagate westward and are slow (low-frequency) and long. These waves play an important role in the transfer of energy in the ocean and atmosphere. For the equatorial \( \beta \)-plane approximation \( f = \beta y \), Rossby waves are exact solutions of linear SWEs. These solutions are in a steady state in a moving frame. When these solutions are considered as the initial condition, they must be preserved by an ideal numerical method. However, most well-known schemes fail in preserving these solutions. In this section, we give some details about the analytical solution of SWEs corresponding to the symmetric equatorial Rossby waves of index 1, which will be used as the initial condition to test the proposed method.

We consider a Rossby wave with wavelength \( L_w = 5500 \text{ km} \) and a domain \([0, 2L_w] \times [0, L_w]\). The mean depth and the reduced gravity
are taken as $H = 300$ m and $g = 3 \times 10^{-2}$ m s$^{-2}$, respectively. Using the characteristic depth, length, time, and velocity values given by Eq. (4) in Section 2.1, we obtain a dimensionless domain $[0,2L] \times [0,1]$ with $L = 16$ and a dimensionless wavelength $X = 16$. In the following, for simplicity, we drop the $\sim$ sign for dimensionless parameters.

The analytical solution is given by:

$$
\begin{align*}
\nu(x,y,t) &= -y \cos(kx-ot)e^{-\gamma t/2} \\
r^+(x,y,t) &= r^+ \frac{1}{kx+\eta} \sin(kx-ot)e^{-\gamma t/2} \\
r^-(x,y,t) &= r^- \frac{1}{kx-\eta} \sin(kx-ot)e^{-\gamma t/2} \\
\eta(x,y,t) &= \left(\frac{r^+}{r^-}\right)^{1/2} \\
u(x,y,t) &= -r^+(x,y,t) \cos(kx-ot)e^{-\gamma t/2}
\end{align*}
$$

(38)

where $r^+ = u - \eta$ and $r^- = u + \eta$ are the Riemann invariants, $k = 2\pi/X$ is the wavenumber, and $\omega$ is the smallest root in magnitude of the dispersion relation:

$$
\omega^2 - k^2 - \frac{k}{\omega} = 3.
$$

The dispersion relation (39) for $X = 16$ leads to $\omega = -0.12512089$. The dimensionless wave period is thus $\bar{T} = 50$, which corresponds to a period of 70 days in a real scale. The wave amplitude can be selected by setting the value of the constant $r$. Fig. 3 shows the analytical form of the Rossby waves ($\eta/r$) at time $t = 500$ (23 months).

### 4.2. Performance of the proposed method

#### 4.2.1. Analysis of the accuracy

In this section, the analytical solution of SWEs corresponding to the symmetric equatorial Rossby waves of Index 1 is used as the initial condition to test the proposed method. The results are compared with those of the $\kappa$ schemes using a triangular mesh with a cell area of 1100 km$^2$. When $\kappa$ schemes are combined with a fourth-order Adams method as a temporal integration scheme, the results are not accurate for Rossby waves, and a high level of numerical diffusion is observed. Fig. 4 shows the water surface elevation using the Quick scheme, which is the best one among $\kappa$ schemes, for Rossby waves at time $t = 250$ (five periods). A high level of damping is already observed at this simulation time, which shows that the $\kappa$ schemes are not accurate in the modeling of Rossby waves.

Figs. 5 and 6 show the water surface elevation using the proposed method for Rossby waves at time $t = 500$. A great improvement is observed compared with Fig. 4, while the symmetric form of the flow is still well preserved. The analytical solutions of water surface elevation for the Rossby waves and water surface elevation using other upwind methods are also shown in Fig. 6. As can be observed in those figures, while the $\kappa$ schemes have highly damped the waves, the proposed method leads to accurate results for Rossby waves, and the damping and phase errors are negligible.

We will use the algorithm used by Bona et al. [3] to obtain the spatial and temporal orders of accuracy. To determine numerically the spatial convergence rate we use very small time step $\Delta t$ in order to render the temporal errors negligible. Different sizes of the computational cells are used to obtain the numerical value of the spatial order of the proposed method. To measure the errors we will use the mesh-size weighted $L^2$-norm, denoted by $\| \cdot \|_2$. Fig. 7 shows the $L^2$ error in log–log scale for the test of Rossby waves at time $t = 250$, and we obtain an order of spatial accuracy of 2.74. The combined effects of spatial and temporal errors are in general difficult to distinguish. For a fixed size of mesh $\Delta x = \sqrt{\Delta t}$, we consider a reference solution at time $t$ which can be obtained by using a small time step $\Delta t_{ref}$. This reference solution will differ from the exact solution by an error that is almost purely from the spatial discretization. This solution is used to cancel the spatial errors. Then for a fixed spatial size $\Delta x$ of the computational cell, we define a modified error at time $t$ denoted by $E'(t)$ associated to the values of $\Delta t$ that are larger than $\Delta t_{ref}$, as follows:

$$
E'(t) = \| h^{(n)}(\Delta x, \Delta t) - h^{(n)}_{ref}(\Delta x, \Delta t_{ref}) \|_2 / \| h^{(o)} \|_2
$$

where $h^{(n)}$ and $h^{(o)}$ are respectively the numerical and the analytical water depth at time $t$ obtained by using the spatial step $\Delta x$ and the temporal step $\Delta t$. The reference solution $h^{(n)}_{ref}$ is the numerical water depth at the same time $t$ which is obtained by using the reference time step $\Delta t_{ref}$ and the same spatial step $\Delta x$. For a small values of $\Delta t$ which are larger than $\Delta t_{ref}$, the temporal rate of convergence can be visible because when we subtract the reference solution $h^{(n)}_{ref}(\Delta x, \Delta t_{ref})$ from the approximate solution $h^{(n)}(\Delta x, \Delta t)$, the spatial errors are almost canceled. The temporal rate of convergence at time $t = 250$ is shown in Table 1 for $\Delta t_{ref} = 0.001$. The results confirm that the splitting method has less impact on the fourth order of the accuracy of the Adams method used as temporal scheme.

![Fig. 3. Water surface elevation for Rossby waves: the analytical form at time $t = 500$.](image1)

![Fig. 4. Water surface elevation for Rossby waves using the fourth-order Adams method and Quick scheme at time $t = 250$ with CFL = 0.1.](image2)
Symmetric equatorial Rossby waves of Index 1, used in this section, are harder to capture by the numerical methods compared to Kelvin and Yanai waves commonly used in the numerical tests for well-balanced property [15]. Following the accuracy of the proposed method for this type of waves, we conclude that the balance between the flux and Coriolis terms is preserved.

4.2.2. Energy conservation

We now study the behavior of the proposed method and \( \kappa \) schemes with respect to numerical energy dissipation. Since the waves used in this section are steady state in a moving frame, they should be preserved by the numerical methods. In particular, their kinetic energy should be also preserved. Fig. 8 shows the change in kinetic energy up to time \( t = 500 \) using the proposed method and \( \kappa \) schemes. The upwind-centered scheme shows large oscillations in kinetic energy. For the other \( \kappa \) schemes the energy decreases rapidly. At time \( t = 250 \), the energy of the Fromm scheme is reduced to 50%, while for the proposed method at this time, the rate of energy dissipation is negligible (less than 1%). Therefore, the proposed method performs very well in the conservation of kinetic energy, even for long simulation times. This is of crucial importance in the simulation of large-scale oceanic and atmospheric flows, where Rossby waves play an essential role in the transfer of energy.

5. Numerical experiments for nonlinear cases

In this section, in order to validate the performance of the proposed method for nonlinear SWEs, some nonlinear test cases are presented. The tests are performed for nonlinear gravity waves, parabolic flood waves and nonlinear Rossby soliton waves. Finally,
the effect of the mesh structure on the solution quality is analyzed using different grids.

5.1. Nonlinear gravity waves

Here, nonlinear gravity waves are considered in order to test the proposed method. We consider a non-dimensional domain $[-L/2, L/2] \times [-L/2, L/2]$, with $L = 150$. A Gaussian distribution of water surface elevation is assumed as the initial condition for the dimensionless form of the nonlinear SWEs (1) and (5) without Coriolis effect.

$$\eta(x, y, 0) = 0.05 e^{-0.1(x^2+y^2)}$$

$u(x, y, 0) = 0,$

$v(x, y, 0) = 0$

Fig. 9 shows the solution for the proposed method at time $t = 50$ with $\text{CFL} = 0.4$ and cell area $\Omega_m = 2.25$. Since no explicit exact solution is available for this case, we use the third-order upwind scheme combined with the TVDRK3 method and a fine mesh with cell area $\Omega_m = 1$ to compute a reference solution. This scheme is chosen in order to obtain a reference solution for gravity waves only, since it performs well for this type of wave, but it leads to a high level of damping for Rossby waves, as shown in Section 4 for all k-schemes. As seen in Figs. 9 and 10, the proposed method leads to good results; the solution remains symmetric, free of numerical oscillations, and accurate over the entire domain.

5.2. Parabolic flood waves

In this section, the proposed method is analyzed using an exact solution which corresponds to time-dependent flows for nonlinear SWEs. The analytical solution given in Thacker [26] for parabolic flood waves is employed. This solution is written as:

$$u(x, y, t) = \frac{x t}{t^2 + T^2}$$

$$v(x, y, t) = \frac{y t}{t^2 + T^2}$$

$$h(x, y, t) = h_0 \left[ \frac{T^2}{t^2 + T^2} - \frac{x^2 + y^2}{R_0} \left( \frac{T^2}{t^2 + T^2} \right)^2 \right]$$

where $h_0$ is the initial height of the peak of the parabolic water surface and the parameters $R_0$ and $T$ satisfy the following equation

$$T = R_0 (2gh_0)^{-1/2}$$

In this test, a non-dimensional domain $[-6,6] \times [-6,6]$ is considered with $h_0 = 1$ and $R_0 = 14$. The initial condition is given as:

$$u(x, y, 0) = v(x, y, 0) = 0.$$  

$$h(x, y, 0) = h_0 \left( 1 - \frac{x^2 + y^2}{R_0} \right)$$

The exact solution is used to calculate the boundary conditions of the domain. Fig. 11 shows the temporal evolution of the numerical solution using the proposed method and the analytical solution along section $y = 0$ using the cell area $\Omega_m = 0.01$. The 3D views of the solution (not shown here) show that the symmetric form of
the flow is still well preserved. Following those tests, we conclude that the proposed method produces stable results while maintaining good accuracy for long time steps until the solution reaches its asymptotic convergence.

5.3. Nonlinear Rossby soliton waves

In this part, we present a test for an equatorial nonlinear Rossby wave. This wave is driven by gravity and rotational forces. Asymptotic solutions of the system are approximated by using Boyd’s previous work on equatorial Rossby soliton waves [5,4]. A nondimensional domain \([-24,24] \times [-8,8]\) is considered in the present test. The boundary conditions used are non-flux along the walls. Eqs. (5) and (6) are considered by using the parameters \(h = H + \eta\) with \(H = 1, g = 1,\) and \(f(y) = y.\) The approximate asymptotic solution is obtained with a first-order expansion of function with an order of 12 for Hermite polynomials, as explained in Boyd [5]. The simulations are performed up to \(t = 40\), with CFL = 0.3 and using the cell area \(X_m = 0.04.\) Even though the asymptotic analytical solution is only a first-order approximation, this solution can be used to check the phase speed of the wave and the conservation of total energy.

Fig. 12 shows the free surface elevation \(\eta\) of the nonlinear Rossby soliton wave in three dimensions for the proposed method at time \(t = 40\). At this time, the analytical solution predicts that the peak of the wave will be \(\eta_{\text{max}} = 0.156\) and for the proposed method the peak is \(\eta_{\text{max}} = 0.154.\) The proposed method preserves the shape of each part of the soliton and gives the correct position and phase speed of propagation of the wave compared to the asymptotic solution of Boyd [5]. For the first part of the soliton \(x\)-position = \(-15.80\) and \(y\)-position = 1.267, and for the second part of the soliton \(x\)-position = \(-15.80\) and \(y\)-position = -1.267.

Finally, Fig. 13 presents the evolution of total energy and shows that the proposed method performs well for the nonlinear Rossby soliton wave in conservation of energy, which is a crucial requirement in atmospheric and oceanic simulations.

5.4. Effect of grid structure

In this section the effect of the mesh structure on the solution quality is analyzed. To characterize the mesh shapes, we choose a skewness parameter defined as:

\[
\text{Skewness} = \frac{\text{Optimal Cell Size - Cell Size}}{\text{Optimal Cell Size}}
\]  

where for a triangle grid the optimal cell size is the size of an equilateral triangle with the same circumscribed circle. The skewness parameter is widely used as an indicator of grid quality (e.g., [23]). For an excellent mesh, the skewness parameter should be in the range values of 0–0.25, and for good and acceptable grids respectively, it should be in the ranges of 0.25–0.50 and 0.50–0.75. In this section, we consider three triangular meshes, shown in Fig. 14, with different grid qualities according to the skewness values. The first case (a) is for an acceptable triangular grid, the second case (b) represents a good mesh, and the third case is an
excellent mesh, based on the skewness parameter. The tests are performed using parabolic flood waves until time \( t = 100 \) to verify the impact of the three meshes on the numerical results. The relative true errors of the water depth for the three cases at time \( t = 100 \) are presented in Table 2. The results demonstrate that for all three cases, the proposed scheme leads to a high level of accuracy and that the best result is obtained for case (c), which has a small skewness coefficient.

6. Conclusion

In this paper, a new numerical scheme using unstructured grids is developed for shallow water flows dominated by rotation effects. This method leads to accurate results for both gravity and Rossby waves, which is of crucial importance in the simulation of large-scale flows. A special treatment of the Coriolis effect was employed in which the Coriolis term is integrated analytically before and after solving the conservation law. This method uses polynomial fitting with high accuracy on both sides of the interface of the computational cells. A fourth-order Adams method with an operator splitting scheme is used for temporal integration. This approach is enough to suppress the short-wave numerical noise without damping the long waves that are essential in the transport of energy Rossby waves in the ocean and atmosphere. The energy conservation of the scheme is considerably improved compared with other upwind schemes, such as the third-order scheme, the cell-averaged scheme, the Fromm method, and the QUICK scheme, which are widely used for simulation of gravity waves and lead to high levels of energy dissipation and numerical oscillation caused by the imbalance between the source and flux terms at the discrete level for Rossby waves. Finally, the employed fourth-order Adams method was found to be an accurate and efficient scheme for the time integration of large-scale shallow water equations, with no iteration on the corrector step being needed to stabilize the method, and optimal results are obtained with only one iteration.

Acknowledgement

The authors thank the anonymous reviewers for their valuable comments and suggestions to improve the quality of the paper.

This publication was made possible by NPRP Grant 4-935-2-354 from the Qatar National Research Fund (a member of the Qatar Foundation). The statements made herein are solely the responsibility of the authors.

Appendix A. The Jacobian matrix for SWEs

A.1. The Jacobian matrix for linear equations

The matrix \( \mathbf{J} \) satisfies \( \Delta \mathbf{F} = \mathbf{J} \mathbf{A} \mathbf{U} \) with:

\[
\mathbf{J} = \frac{\partial (\mathbf{F} \cdot \mathbf{n})}{\partial \mathbf{U}} = \begin{pmatrix} 0 & H_n \eta & H_n \eta \\ g_n & 0 & 0 \\ g_n & 0 & 0 \end{pmatrix}
\]

where \( c = \sqrt{gh} \). The eigenvalues of \( \mathbf{J} \) are given by:

\[
\lambda_1 = c, \quad \lambda_2 = 0, \quad \lambda_3 = -c
\]

with the corresponding eigenvectors

\[
\mathbf{e}_1 = \begin{pmatrix} 1 \\ \lambda_0 R_n \eta \\ -\lambda_0 R_n \eta \end{pmatrix}, \quad \mathbf{e}_2 = \begin{pmatrix} 0 \\ -\lambda_0 R_n \eta \\ \lambda_0 R_n \eta \end{pmatrix}, \quad \mathbf{e}_3 = \begin{pmatrix} 1 \\ -\lambda_0 R_n \eta \\ -\lambda_0 R_n \eta \end{pmatrix}
\]

where

\[
\lambda_0 = \sqrt{\frac{g}{H}}
\]

The coefficients \( \lambda_k, k = 1, 2, 3, \) are computed as:

\[
\begin{align*}
\lambda_1 &= \frac{\lambda_0}{2} + \frac{1}{2}\left(\Delta(hu)w + \Delta(hv)n_x - (\hat{u}n_x + \hat{v}n_y)\Delta h\right) \\
\lambda_2 &= \frac{1}{2}\left((\Delta(hv) - \hat{v}\Delta h)n_y - (\hat{u}n_x + \hat{v}n_y)\Delta h\right) \\
\lambda_3 &= \frac{\lambda_0}{2} - \frac{1}{2}\left((\Delta(hu) + \hat{u}\Delta h)n_x - (\hat{u}n_x + \hat{v}n_y)\Delta h\right)
\end{align*}
\]

A.2. Non-dimensional linear system

The flux and source vectors in the dimensionless form become:

\[
\mathbf{E} = \begin{pmatrix} u \\ \eta \\ 0 \end{pmatrix}, \quad \mathbf{G} = \begin{pmatrix} \nu \\ 0 \\ \eta \end{pmatrix}, \quad \mathbf{S} = \begin{pmatrix} 0 \\ yv \\ -uy \end{pmatrix}
\]

The eigenvalues and eigenvectors of the non-dimensional system can be calculated using those of the original system by simply setting \( H = g = 1 \).

A.3. The Jacobian matrix for nonlinear equations

The Jacobian matrix \( \mathbf{J} \) for the nonlinear case is given by:

\[
\mathbf{J} = \frac{\partial (\mathbf{F} \cdot \mathbf{n})}{\partial \mathbf{U}} = \begin{pmatrix} 0 & n_x & n_y \\ (\dot{\epsilon} - \epsilon n_x - 2\epsilon m_x + 2\epsilon m_y) & n_x & n_y \\ (\epsilon m_y - (\epsilon^2 - \epsilon^2)n_x) & n_x & n_y \end{pmatrix}
\]

with the following eigenvalues and eigenvectors

\[
\begin{align*}
\lambda_1 &= \hat{u}m_y + \hat{v}m_x - \dot{c}, \quad \lambda_2 = \hat{u}n_x + \hat{v}m_y, \quad \lambda_3 = \hat{u}n_x + \hat{v}m_y - \dot{c} \\
\mathbf{e}_1 &= \begin{pmatrix} 1 \\ \hat{u} + \epsilon n_x \\ (\dot{c} - \epsilon n_x) \end{pmatrix}, \quad \mathbf{e}_2 = \begin{pmatrix} 0 \\ -\epsilon m_x \\ \epsilon m_x \end{pmatrix}, \quad \mathbf{e}_3 = \begin{pmatrix} 1 \\ \hat{u} - \epsilon n_x \\ (\dot{c} - \epsilon n_x) \end{pmatrix}
\end{align*}
\]

where

\[
\dot{\epsilon} = \frac{u_k \sqrt{h_k} + u_l \sqrt{h_l}}{\sqrt{h_k} + \sqrt{h_l}}, \quad \dot{c} = \frac{v_k \sqrt{h_k} + v_l \sqrt{h_l}}{\sqrt{h_k} + \sqrt{h_l}}, \quad \frac{\epsilon}{2}
\]

The coefficients \( \lambda_k \) depend on the jumps \( \Delta \ell = (\cdot)_k - (\cdot)_l \) as:

\[
\begin{align*}
\lambda_1 &= \frac{\lambda_0}{2} + \frac{1}{2}\left(\Delta(hu)w + \Delta(hv)n_x - (\hat{u}n_x + \hat{v}n_y)\Delta h\right) \\
\lambda_2 &= \frac{1}{2}\left((\Delta(hv) - \hat{v}\Delta h)n_y - (\hat{u}n_x + \hat{v}n_y)\Delta h\right) \\
\lambda_3 &= \frac{\lambda_0}{2} - \frac{1}{2}\left((\Delta(hu) + \hat{u}\Delta h)n_x - (\hat{u}n_x + \hat{v}n_y)\Delta h\right)
\end{align*}
\]

Appendix B. Time integration methods

The following time integration methods are used to solve an ODE defined by:

\[\frac{d}{dt} \mathbf{U} = \mathbf{J} \mathbf{A} \mathbf{U} \]
\[ \frac{d\mathbf{U}}{dt} = f(\mathbf{U}, t), \quad \mathbf{U}(t_0) = \mathbf{U}^0 \]  

We use \( \mathbf{U}^0 \) to denote the computed approximation to the solution at time \( t_n = n\Delta t \).

**B.1. The fourth-order Adams method**

The fourth-order Adams method uses the fourth-order Adams-Bashforth scheme as predictor:

\[
\mathbf{U}^n = \mathbf{U}^{n-1} + \Delta t \left( \frac{55}{24} \mathbf{f}^{n} - \frac{59}{24} \mathbf{f}^{n-1} + \frac{37}{24} \mathbf{f}^{n-2} - \frac{9}{24} \mathbf{f}^{n-3} \right)
\]

and the fourth-order Adams-Moulton scheme as corrector

\[
\mathbf{U}^{n+1} = \mathbf{U}^n + \Delta t \left( \frac{9}{24} \mathbf{f}^{n+1} - \frac{19}{24} \mathbf{f}^n + \frac{5}{24} \mathbf{f}^{n-1} - \frac{1}{24} \mathbf{f}^{n-2} \right)
\]

where

\[ \mathbf{f}^n = f(\mathbf{U}^n, n\Delta t) \]

**B.2. The fourth-order Runge-Kutta method**

The value \( \mathbf{U}^{n+1} \) is calculated using the following formula:

\[
\mathbf{U}^{n+1} = \mathbf{U}^n + \Delta t \left( k_1 + 2k_2 + 2k_3 + k_4 \right)
\]

where

\[ k_1 = f(\mathbf{U}^n, t_n) \]
\[ k_2 = f(\mathbf{U}^n + \frac{1}{2} \Delta t \mathbf{k}_1, t_n + \frac{1}{2} \Delta t) \]
\[ k_3 = f(\mathbf{U}^n + \frac{1}{2} \Delta t \mathbf{k}_2, t_n + \frac{1}{2} \Delta t) \]
\[ k_4 = f(\mathbf{U}^n + \Delta t \mathbf{k}_3, t_n + \Delta t) \]

**B.3. The third-order TVD Runge–Kutta method**

We consider the following ODE

\[
\frac{d\mathbf{U}}{dt} = \mathbf{L}(\mathbf{U}), \quad \mathbf{U}(t_0) = \mathbf{U}^0
\]

where \( \mathbf{L} \) is a spatial operator. The TVDRK3 method [24] is performed via three stages to solve Eq. (62).

\[
\begin{align*}
\mathbf{U}^{(1)} &= \mathbf{U}^0 + \Delta t \mathbf{L}(\mathbf{U}^0) \\
\mathbf{U}^{(2)} &= \frac{3}{4} \mathbf{U}^0 + \frac{1}{2} \mathbf{U}^{(1)} + \frac{1}{4} \Delta t \mathbf{L}(\mathbf{U}^{(1)}) \\
\mathbf{U}^{n+1} &= \frac{1}{3} \mathbf{U}^{(1)} + \frac{2}{3} \mathbf{U}^{(2)} + \frac{2}{3} \Delta t \mathbf{L}(\mathbf{U}^{(2)})
\end{align*}
\]

**References**


