Improving multi-block sigma-coordinate for 3D simulation of sediment transport and steep slope bed evolution

Zeinab Khorrami, Mohammad Ali Banihashemi*

School of Civil Engineering, University College of Engineering, University of Tehran, P.O. Box 11365-4563, Tehran, Iran

A R T I C L E   I N F O

Article history:
Received 28 November 2017
Revised 16 October 2018
Accepted 22 October 2018
Available online 29 October 2018

Keywords:
Multi-block
Sigma coordinate
Sediment transport
Local mesh refinement
Long-term simulation

A B S T R A C T

This paper aims at developing a multi-block sigma-coordinate to simulate morphological evolutions. The developed multi-block sigma-coordinate can represent a steep slope topography smoothly with different numbers of horizontal layers, without producing any truncation error and artificial flux (PGFE). The multi-block sigma-coordinate can easily increase the depth-direction resolution of the sediment transport module in the sub-regions, without the aggregation of computational points in the shallow areas which may be caused by using high resolution over the entire domain. The model is beneficial for long-term simulation of morphological evaluations in lakes where the bed slope near delta region (the sedimentary area which forms where a river enters a lake/ocean) is mostly steep, and delta keeps advancing down the lake.

The multi-block sigma approach at the block interface, where the number of horizontal layers varies, allocates the flux to the neighboring cells according to two essential factors, namely “the common border length” and “satisfying continuity” which lead to defining the virtual cells. A series of numerical tests have been performed. Comparison between the numerical results, the analytical solutions, and the experimental data demonstrated an appreciable accuracy, the satisfactory performance and the efficiency of this scheme.

© 2018 Elsevier Inc. All rights reserved.

1. Introduction

The three-dimensional and the laterally averaged (2D-Vertical) transport models are applied to simulate sediment transport, water quality and bed evolutions in irregular and steep slope bathymetries. As to the vertical coordinate and for free surface flow, some of these models solve the governing equations in Cartesian coordinates system, well-known as the z-level, and the others use a transformed coordinate, the so-called sigma-coordinate as introduced by Philips (1957).

A z-level model slices the water column into horizontal layers with a constant thickness, while the number of computational points and the vertical spacing (dz) can vary in the z-direction (Fig. 1(a)). While a sigma-coordinate divides the total water column into a constant number of layers. For each layer, the thickness can vary in the depth/z direction, but the ratio of the layer thickness to the water depth is constant [1,2]. There are benefits and drawbacks in both systems. The z-level model cannot properly fit the bed topography and needs some remedies such as using partial cells and a piecewise linear bottom for a better approximation of the bed [3–10]. Though a sigma-coordinate model does not have this topography-fitting problem, it produces artificial flux in the steep slope bottom areas [1,9,11–13].

* Corresponding author.

E-mail addresses: Zkhorr@UT.Civil.ac.IR (Z. Khorrami), Banihash@UT.Civil.ac.IR (M.A. Banihashemi).

https://doi.org/10.1016/j.apm.2018.10.027
0307-904X/© 2018 Elsevier Inc. All rights reserved.
Transforming to the curvilinear coordinate ($x^*, y^*, \sigma$) is done by applying the chain rule to the derivative terms of the Cartesian coordinates equations ($x, y, z$), ($x^* = x, y^* = y, \sigma = (z - \zeta)/H$). So, in the new curvilinear coordinate, the horizontal gradient of a parameter near the steep slope bottom might be the sum of two relatively large terms with opposite signs which tend to eliminate each other.\[
\frac{\partial \phi}{\partial x^*} = \frac{\partial \phi}{\partial x} \frac{\partial x}{\partial x^*} + \frac{\partial \phi}{\partial y} \frac{\partial y}{\partial x^*} - \frac{\partial \phi}{\partial z} \frac{\partial z}{\partial x^*} - \frac{1}{H} \left( \frac{\partial H}{\partial x^*} + \sigma \frac{\partial H}{\partial \sigma} \right) \frac{\partial \phi}{\partial \sigma}.
\]Small truncation errors in approximating the two terms, can result in a relatively large error, and the horizontal gradient of the parameter ($\partial \phi/\partial x$) might be estimated to be non-zero, even if it is zero and this produces an artificial flux [1]. By solving 3D or 2D-Vertical flow equations in the sigma-coordinate near the steep slope bottom, this error can occur in the horizontal pressure gradient forces of the momentum equation and thus is called Horizontal Pressure Gradient Force Error (PGFE) and increases the numerical diffusions [9,13–16].

Recent improvements to reduce PGFE have been done by increasing the order of truncation errors; for instance, using fourth-order, sixth-order, and second-and-sixth-order approximations [4,17,18]. This method increases the computational costs and is not proper for simulating larger scales [17]. Another method is the subtraction of the reference density which deviates the pressure gradient from a chosen reference density [9,18–22,14,15]. This method has been applied for simulating several areas such as the Atlantic Ocean [14], the Nordic Sea [15] and the Hardanger fjord [9], is useful for modeling small areas, where the differences in horizontal density are small within the computational domain; hence, it is unprofitable in large areas [20]. The topography-smoothing method is a new technique. It reduces the horizontal depth gradient using smoothing the physical topography bottom through introducing a flat virtual bottom [16,23,24], and it is used for flow simulation in bends [23] and Adriatic Sea [24]. This method is needed to identify boundary cells and to track them. The
topography smoothing method creates an additional source of errors [16]. Discretizing the domain using sigma-coordinate, but discretizing the governing equations along Cartesian coordinates \((x, y, z)\), seems to be one of the most acceptable methods for steep slope problems. In this method, discretizing a term such as \((\partial C/\partial x)_h\) is based on the values of the neighboring cells at the \(k\) equivalent level \((C_{i-1/2, j+1/2, k})\) which should be calculated by interpolating \(C\) values at \(k-I, k, k+I\) points. Choosing \(C_{i-1/2, j+1/2, k}\) may vary in different ways and is important for the accuracy and the conservation of this method [14,20,25–27]. The second-order Lagrangian interpolation polynomial upon the finite difference method [12], so-called weighted Jacobian method [25], the level of intersection of diagonals [14, 20], and the approximation of the diffusion flux and pressure gradient based on a finite volume method [1], have been used to discretize the equation in Cartesian coordinates.

In this paper, the discretization is based on the average value of \(C\) at the edges of a cell control volume; it can be calculated from the cell center value and the center value of four surrounding cells [28]. The truncation error and the artificial flux near the steep slope bottom will not exist by applying this method.

Using the sigma-coordinate at the steep slope bottom is prone to another difficulty caused by dividing the depth into a constant number of layers. The constant number of layers causes the aggregation of computational points in the shallow depth areas; which can constrain additional computational costs. To create a local mesh refinement, different methods may be applied, such as patching method \((zonal \ grid)\), unstructured mesh, and nested grid. A mesh patching method decomposes the computational domain by using a main grid and a set of sub-grids with different resolutions; it can be attached or embedded to/in the main grid [29–32]. The individual mesh components of the main grid and sub-grids are hydro-dynamically linked [33], or connected at the patching interface by the numerical flux [34]. A locally refined mesh can be created by applying an adaptive unstructured grid, generated for the whole domain with increased resolution in a region of interest, and without any requirement for using the sub-grid [35–37]. However, using the unstructured grid can be limited due to its complex creation techniques and the necessity of the massive storage for grid data. In a nesting method, a fine grid is embedded in a coarse one, where the two grids are dynamically linked through boundary conditions by using one-way or two-way interacting schemes [38–41]. In the two-way approach, the information is exchanged between the solutions of the coarse grid and the fine grid, while in the one-way scheme, only the coarse grid can have the impact on the fine grid solution. The horizontal nested grid is widely used in modeling shallow areas, where the horizontal distances are much larger than the vertical depth. However, in recent years, vertically nested modeling is applied by Weather Research and Forecasting Models [42,43]; but, nested modeling and other approaches are constrained by the lack of using vertical grid refinement, especially in hydrodynamic and sediment transport simulation.

This paper developed a multi-block grid in the sigma-coordinate for the first time to increase local grid refinement and to optimize the computational points with different numbers of horizontal layers. Another modified sigma-coordinate system had been developed earlier as well, where non-equal vertical intervals of the sigma-coordinate had been modified, whereas any change in the water elevation can result in a change in the grid size over all of the vertical intervals [44]. The new improved multi-block sigma-coordinate represents the steep slope smoothly and can be useful for the long-term morphological modeling of lakes where the bed slope is mostly steep near delta regions which are sedimentary formation at the entrance of lakes, and keeps advancing toward down the lake. By raising the bed level and decreasing the depth, the computational point will decrease in delta regions.

This paper is organized as follows. The first section gives all governing equations and boundary conditions. In the second section, the numerical method applied to solve the equations is determined. The third section, gives a detailed explanation of the multi-block sigma-coordinate and the schemes for solving ‘Advection’ and ‘Diffusion’ terms in the two sub-sections. In the fourth section, the accuracy and the efficiency of the scheme are investigated. Some conclusions are drawn in the final section.

2. Mathematical model

2.1. 2D shallow water flow equations

Based on the relatively shallow depth of reservoirs, as compared to their surface area, the governing equations applied to describe the flows are the shallow water equations, which assume that the pressure is hydrostatic:

\[
\frac{\partial h}{\partial t} + \frac{\partial (U h)}{\partial x} + \frac{\partial (V h)}{\partial y} = 0
\]

\[
\frac{\partial (hU)}{\partial t} + \frac{\partial (hU^2)}{\partial x} + \frac{\partial (hUV)}{\partial y} + g h \frac{\partial \eta}{\partial x} = \frac{\partial}{\partial x} \left( \nu_t \frac{\partial (hU)}{\partial x} \right) + \frac{\partial}{\partial y} \left( \nu_t \frac{\partial (hU)}{\partial y} \right) + \frac{1}{\rho} \tau_{bx}
\]

\[
\frac{\partial (hV)}{\partial t} + \frac{\partial (hUV)}{\partial x} + \frac{\partial (hV^2)}{\partial y} + g h \frac{\partial \eta}{\partial y} = \frac{\partial}{\partial x} \left( \nu_t \frac{\partial (hV)}{\partial x} \right) + \frac{\partial}{\partial y} \left( \nu_t \frac{\partial (hV)}{\partial y} \right) + \frac{1}{\rho} \tau_{by}
\]

where \(\eta = h + z_s\), and \(h\) and \(\eta\) refer to the water depth and water surface elevation, respectively; \(Z_b\) is the bed level; \(U\) and \(V\) are the components of the depth-averaged velocities in the \(x\) and \(y\) directions, respectively; \(\nu_t\) is the eddy viscosity; \(g\) is the gravitational acceleration; \(\rho\) is fluid density; \(t\) is time; \(\tau_{bx}\) and \(\tau_{by}\) by refer to the bed shear stresses in the \(x\) and \(y\)
directions, respectively, which are computed as \( \tau_{by} = \rho C_f V \sqrt{U^2 + V^2} \), \( \tau_{bx} = \rho C_f U \sqrt{U^2 + V^2} \), \( C_f = \frac{8}{3n^2} \), where \( n \) and \( C \) are the Manning and Chezy roughness coefficients, respectively.

2.2. 3D sediment transport equations

Computing the sediment transport is divided into the suspended and bedload transport, depending on the size and density of the bed material as well as the flow conditions. The suspended sediment transport is computed by solving the advection-diffusion equation. The 3D mass equation for the sediment transport in suspension is as follows:

\[
\frac{\partial c}{\partial t} + \frac{\partial (u c)}{\partial x} + \frac{\partial (v c)}{\partial y} + \frac{\partial (w c)}{\partial z} = \frac{\partial}{\partial x} \left( \varepsilon_x \frac{\partial c}{\partial x} \right) + \frac{\partial}{\partial y} \left( \varepsilon_y \frac{\partial c}{\partial y} \right) + \frac{\partial}{\partial z} \left( \varepsilon_z \frac{\partial c}{\partial z} \right) + S
\]

where \( C \) is the suspended sediment concentration; \( u \) and \( v \) are the local fluid velocities at elevation \( z \) in the \( x \) and \( y \) directions, respectively; \( w \) is the fall velocity of the sediment particles; \( \varepsilon_i \) is the turbulent diffusivity in \( i \) direction (where \( i=x, y, z \)), and \( S \) refers to the sediment source term.

The exchange of sediment between the interface of the suspended and bed-load layers, with the thickness \( h \), is through the suspended sediment deposition, \( D_h = w_s C_s z - b \) and the bed-load entrainment, \( E_b \). Hence, the net flux across the boundary is \( D_h - E_b \), and the term appears in the bed-load mass balance Eq. (6) below. Here, \( C_b \) and \( C_b^* \) are concentration and equilibrium concentration at the top of the bed-load layer thickness, and can be calculated by the expressions of van Rijn (1987),

\[
C_b = 0.015 \frac{d_{50} T_1^{1.5}}{b D_e^3} ; \quad C_b^* = C_1 + C_{b0} \left[ 1 - e^{-\left( w_s C_s z / h \right)} \right]
\]

where \( C_1 \) and \( z_1 \) are the concentration value at the neighboring grid point and its location in the \( z \)-direction, respectively, and \( \sigma_c \) is related to the turbulent diffusivity of the sediment and is set 1.0 in the calculations.

2.3. Bed evolution equation

The time evolution of the bed level is obtained through the mass-balance equation for the bed-load layer [47]:

\[
(1 - p) \frac{\partial z_b}{\partial t} + \frac{\partial q_{bx}}{\partial x} + \frac{\partial q_{by}}{\partial y} + D_b - E_b = 0
\]

where \( p \) is the porosity of the bed material, \( q_{bx} \) and \( q_b \) refer to the components of the bedload transport along the \( x \) and \( y \) directions, respectively, which are related to the bed-load \( q_b \) by the direction cosines.

2.4. Auxiliary equations

To solve the 3D sediment transport equation, the velocities distributions along the vertical direction \( (u, v) \) are obtained by applying a logarithmic velocity profile.

\[
u = \frac{U}{h} \ln \left( \frac{h}{z} \right) ; \quad v = \frac{V}{h} \ln \left( \frac{h}{z} \right)
\]

To evaluate the turbulent diffusivity, an algebraic expression (zero-equation turbulence model) is used.

\[
\varepsilon_{x,y} = 0.11 u_* \nu ; \quad \varepsilon_z = \varepsilon_{\text{max}} - \varepsilon_{\text{max}} (1 - 2z/h)^2 \quad \text{for } z < 0.5h ; \quad \varepsilon_z = \varepsilon_{\text{max}} \quad \text{for } z > 0.5h
\]

where \( u_* \) is the shear velocity and \( \varepsilon_{\text{max}} = 0.25h u_* \). Meanwhile, the bed-load transport is calculated using an empirical formula introduced by van Rijn [53],

\[
q_b = 0.053 \left( \frac{\rho_s - \rho}{\rho} \right)^{0.5} \frac{d_{50} 1.5 T_1^{2.1}}{D_e^{3.1}}
\]

where \( q_b \) is the transport rate of the bed-load per unit width, \( \rho_s \) is the density of the sediment particles; \( d_{50} \) is the median diameter of the sediment particles, \( D_s \) is the diameter parameter, \( D_s = d_{50} \left( \frac{\rho_s - \rho}{\rho} \right)^{0.5} \); \( \nu \) is the kinematic viscosity of the fluid, and \( T \) is the transport stage parameter, \( T = (u^2 - u_{cr}^2)/u_{cr}^2 \), \( u_{cr} \) is the critical shear velocity.

2.5. Boundary conditions

The boundary conditions for the inlet of the domain can be “known boundary conditions”, implementing the known values of velocities, discharge, or water surface elevations; and, for the outlet, an “open boundary condition” can be used. In the “open boundary condition”, the derivatives of the parameters are set to zero and the outgoing waves pass over the exterior of a computational domain without creating any reflections on the boundary. At the solid walls, the no-slip condition is considered where the normal velocity component is zero at the boundary.
3. Numerical method

The 2D shallow water equations (2DH) are calculated in Cartesian coordinates, and a sigma–coordinate is used for the sediment transport equation. A rectilinear grid system is formed by mesh projection on the horizontal plane. The Cartesian-cell method uses a stair-cased surface representation of curvilinear boundaries, as opposed to a cut-cell or immersed-boundary treatment. The main variables are defined in the staggered grid system; so, the water level and concentration are considered to be located on the cell center indicated by \((i, j, k)\), and the velocity components \((u, v, w_i)\) are placed at the faces \(i = \pm 1/2, j, k\), \((i, j = \pm 1/2, k)\), \((i, j, k = \pm 1/2)\), respectively (Fig. 1(b)).

The finite volume method is used to discretize the governing equations. A fractional step algorithm (time splitting) is applied to solve different terms of the governing equations. This scheme solves the terms of the governing equation in some steps using the known intermediate values calculated in the previous step. The second–order accurate Fromm [48] method is used for ‘Advection terms’, and the Alternating Direction Implicit (ADI) scheme which is the second–order accurate in time and space [49], is applied for ‘Diffusion terms’.

4. Multi-block sigma-coordinate

The multi-block sigma-coordinate can be adjusted to the morphological bed changes. To this aim, the computational domain is first divided into a number of blocks (the sub-domains). The number of horizontal layers is constant within each block but it changes from block to another. In this case, one or more threshold heights are introduced. For heights less than the threshold, \(h < H_{\text{threshold}}\), the block with less horizontal layers, and for heights greater than the threshold, \(h > H_{\text{threshold}}\), the block with more horizontal layers are applied. The number of horizontal layers is chosen arbitrarily. Fig. 1 shows a typical multi-block sigma-coordinate with two blocks which have \(n\) and \(m\) horizontal layers \((\text{where} \; n \neq m)\), as applied to a lake.

Using the multi-block sigma-coordinate, no considerable effect on the time step applied to the computations defined within the application range. The numerical stability criteria link \(dt\) (the time step) to \(dx\) and \(dy\) (the grid distances), by using the Courant Number. The time step \((dt)\) is dictated by the grid distances and velocity components. Since the fall velocity of sediment particles \((w_i)\), is typically much smaller than the components of flow velocity, it does not determine the time step. When carrying out a multi-block sigma-coordinate computation, the scheme in each of the blocks is the same as if a constant number of horizontal layers is used throughout the entire model domain, except a narrow buffer zone at the block interface. In the buffer zone, the virtual cells are defined so that information can be transferred accurately. Returning to the previous section, the transport equation consists of advection and diffusion terms, whereas it should be solved in the multi-block sigma-coordinate.

4.1. Advection term

At the block interface, the number of neighboring cells varies at one of the two vertical sides of the computational cell, and it needs more efforts for discretizing. In the computer program, first, all of the neighboring cells of the computational cell should be determined; then, the horizontal gradient of concentration \((\partial C/\partial x)\), is calculated. Determining the neighboring cells, which have the common boundary with the computational cell at the block interface, is according to cell locations defined by using \(\Delta z_n, \Delta z_m, d_{i, j, k}, d_{i + 1/2, j, k}\), as shown in Fig. 2. The concentration horizontal gradient, \(\partial C/\partial x\), is calculated through the second-order Fromm method and the average value of \(C\) at the edges of a control volume,

\[
\frac{\partial C}{\partial x} \approx \sum_{j=1}^{nj} C_j \frac{\Delta z_j}{Ar}
\]

(10)

where, \(nj\) is the number of control volume edges, \(C_j\) is the average values of the concentration on the \(j\)th edge, \(Ar\) is the control volume area, and \(\Delta z_j\) refers to the projected length of the \(j\)th edge along the \(z\)-direction. For a cell next to the block interface, the \(x\) derivation of concentration at the cell center can be written as,

\[
\left(\frac{\partial C}{\partial x}\right)_{i,k} = 0.5 \times \left(\frac{\partial C}{\partial x}_{i-1/2,k} + \frac{\partial C}{\partial x}_{i+1/2,k}\right)
\]

\[
\left(\frac{\partial C}{\partial x}\right)_{i+1/2,k} = \frac{C_{i+1/2,k-1/2}(z_A - z_E) + C_{i,k}(z_E - z_C) + C_{i+1/2,k+1/2}(z_C - z_D) + \sum_{l=1}^{l2} C_{i+1,k+l}\alpha_l(z_{E} - z_{A})}{Ar}
\]

\[
\left(\frac{\partial C}{\partial x}\right)_{i-1/2,k} = \frac{C_{i-1/2,k-1/2}(z_B - z_E) + C_{i-1,k}(z_E - z_F) + C_{i-1/2,k+1/2}(z_F - z_C) + C_{i,k}(z_C - z_B)}{Ar}
\]

(11)

where, \(C_{i, k \pm 1/2, \pm 1/2}\) refer to concentrations at the control volume edges and are calculated from the average concentration of the four or more surrounding cells. \(l1\) and \(l2\) define the neighboring cells referenced from \(k, l1\) is the number of neighboring cells at the block interface \((l1 = l2 - l1 + 1)\), \(\alpha_l\) denotes the contribution of the neighboring cells depending on the
common border length, \( \alpha_l = (z_k'^{l+1} - z_k'^{l})/(z_k'^{l} - z_k'^{l-1}) \) and \( \sum_{l=1}^{12} \alpha_l = 1.0. \ z_k'^{l+1}, z_k'^{l} \) are the top and bottom levels of the common border of the \( l \)th neighboring cell in the \( z \)-direction, Fig. 2.

Concentration horizontal gradient at cell boundaries, can be rewritten as Eq. (12) to make it more applicable, whereas the first column of the adjacent block (block No. 2) is considered as the last virtual column of the computational cell block (block No. 1). A virtual column means that the first column of the adjacent block with \( m \) cells in the vertical direction, is re-divided into \( n \) cells, similar to the block of the computational cell. A virtual cell concentration is calculated by using the weighted average of the concentration of the initial real cells (\( m \) vertical cells), constructing a virtual cell; and the weights are the contributions of their area to construct a virtual cell, as shown in Fig. 3.
\[
\left( \frac{\partial C}{\partial x} \right)_{i+1/2,k} = \frac{2}{4\Delta x} \left[ \frac{1}{2} \sum_{j=1}^{3} r_{j,1/2}^{i+1/2,k} C_{i+1/2,k-1/2}^{m_{i+1/2,k}}(j), n_{i+1/2,k}(j) \right]
\]

\[
m_{i+1/2,k}(1) = i, m_{i+1/2,k}(2) = i + 1, n_{i+1/2,k}(1) = k - 1, n_{i+1/2,k}(2) = k, n_{i+1/2,k}(3) = k + 1
\]

\[
r_{i+1/2,k} = \frac{1}{4\Delta x} \left[ \Delta z_{i+1/2,k-1/2}^{i+1/2,k} - 4\Delta z_{i+1/2,k}^{i+1/2,k-1/2} - \Delta z_{i+1/2,k-1/2}^{i+1/2,k+1/2} - \Delta z_{i+1/2,k+1/2}^{i+1/2,k-1/2} \right]
\]

At the block interface, the above equation is valid; for the last computational cell, it can be written as:

\[
C_{i+1/2,k}^{m_{i+1/2,k}}(1), n_{i+1/2,k}(2) = \frac{\sum_{j=1}^{3} w_{l,j}^{i+1/2,k} C_{i+1/2,k+j}^{m_{i+1/2,k}}(j), n_{i+1/2,k}(j)}{\sum_{j=1}^{3} w_{l,j}^{i+1/2,k} C_{i+1/2,k+j}^{m_{i+1/2,k}}(j), n_{i+1/2,k}(j)} = 1.0
\]

\[
\Delta z_{i+1/2,k}^{i+1/2,k} - \Delta z_{i+1/2,k}^{i+1/2,k-1/2} = -\frac{\Delta z_{i+1/2,k}^{i+1/2,k-1/2} - \Delta z_{i+1/2,k}^{i+1/2,k+1/2} - \Delta z_{i+1/2,k}^{i+1/2,k-1/2}}{4} = \Delta z_{i+1/2,k}^{i+1/2,k} + \Delta z_{i+1/2,k}^{i+1/2,k-1/2} - \Delta z_{i+1/2,k}^{i+1/2,k-1/2} - \Delta z_{i+1/2,k}^{i+1/2,k-1/2}
\]

where \( n \) is the number of horizontal layers of the block to which the computational cell belongs. \( (\cdot) \) is the data of virtual cells, \( m_{i+1/2,k}(2) \) refers to virtual cells considered as the last cells of the block, \( C_{i+1/2,k}^{m_{i+1/2,k}}(n_{i+1/2,k}(2)) \) is the concentration of the virtual cells calculated by using the concentration of the initial constructor cells, \( w_{l,j}^{i+1/2,k} \) is the weight of the initial constructor cells, \( Area^{n_{i+1/2,k}}_{i+1/2,k} \) is the area of the virtual cell, and \( Area^{n_{i+1/2,k}}_{i+1/2,k} \) presents the area contribution of the initial constructor cells, and \( \Delta z_{i+1/2,k}^{i+1/2,k} \) and \( \Delta z_{i+1/2,k}^{i+1/2,k} \) are the down and up \( z \) levels of the \( k \)th virtual cell at \( X_{i+1} \), respectively.

At the block interface, allocating flux to the neighboring cells is proportional to two essential factors: the common border length, and satisfying continuity relation. The contribution of each neighboring cell to the flux is determined according to the common boundary length of that neighboring cell with the computational cell. Satisfying the continuity relation means that the output (input) flux of a cell must be exactly equal to the sum of the input (output) fluxes of its neighboring cells. At the block interface, the velocity for each virtual interface is calculated from the logarithmic velocity profile Eq. (4) according to the depth-averaged velocities, \( F_{i+1/2,k,l} \) and \( F_{i+1/2,k,l} \) can be written as,

\[
F_{i+1/2,k,l} = u_{i+1/2,k,l} \left( C_{i+1/2,k,l} \Delta z_{i+1/2,k,l} + \frac{\partial C}{\partial z} \right)_{i+1/2,k,l}, l = 1, \ldots, l_{k}, l_{k} = 1, \sum_{l=1}^{k} \Delta z_{i+1/2,k,l} = \Delta z_{i+1/2,k}
\]

\[
F_{i+1/2,k,l} = \begin{cases} \frac{u_{l+1/2,k,l} \left( C_{l+1/2,k,l} \Delta z_{l+1/2,k,l} + \frac{\partial C}{\partial z} \right)_{l+1/2,k,l} \Delta z_{l+1/2,k,l}}{l_{k}} & \text{if } S_{l+1/2,k,l} > 0 \\ \frac{u_{l+1/2,k,l} \left( C_{l+1/2,k,l} \Delta z_{l+1/2,k,l} + \frac{\partial C}{\partial z} \right)_{l+1/2,k,l} \Delta z_{l+1/2,k,l}}{l_{k}} & \text{if } S_{l+1/2,k,l} \leq 0 \end{cases}
\]

where \( l_{k} \) is the number of the neighboring cells and if \( l_{k} = 1 \) the cell is not located at the block interface; also, \( l_{k} \) means that the cell is located at the block interface and has \( l_{k} \) neighboring cells. \( \Delta z_{i+1/2,k} \) is the \( z \)-directional length of the control volume edge and \( \Delta z_{i+1/2,k,l} \) is the common border length of neighboring cells. The flux of the other control volume faces \( (F_{i-1/2,k,l} \) and \( F_{i-1/2,k,l} \) can be calculated by replacing \( k \) with \( k-1 \) and \( i \) with \( i-1 \). The amounts of the fluxes, which are subtracted from the computational cells, are added to the neighboring cells.

To calculate \( \frac{\partial C}{\partial x} \), whereas the cell is from the new computational block (block2), the last column of the previous block (block1) needs to be considered as the first virtual column. The virtual column will be re-divided into \( m \) number cells, same as the new computational block. Calculation of the virtual cells concentration and the other steps are similar to the defined technique, Fig. 3.

4.2. Diffusion term

The diffusion terms are calculated implicitly to specify the second intermediate \( C^{(2)} \) value based on the previous intermediate concentrations achieved through the advection terms. To calculate \( \frac{\partial C}{\partial x} \) for the diffusion (Eq. 14), \( C_{i+1/2,k}^{n} \) and \( C_{i+1/2,k}^{n-1} \) are determined from time \( n + 1/2 \), and the concentrations of the other surrounding cells which take part in calculating \( \frac{\partial C}{\partial x} \) are determined at time step number \( n \). The Alternating Direction Implicit (ADI) method is applied, which leads to the solution of a tridiagonal matrix. This approach is suitable for all cells of a row of a block except for the first and last cells, which are located at the block interface. At the block interface, \( \frac{\partial C}{\partial x} \) is determined from time \( n \).

\[
\frac{\partial C}{\partial x} = \frac{D_{c,v}}{dx} \left( \frac{\partial C}{\partial x} \right)_{i+1/2,k} = \frac{D_{c,v}^{n+1/2,k} \left( \frac{\partial C}{\partial x} \right)_{i+1/2,k}^{n+1/2,k}}{dx}
\]

\[
\frac{(\partial C)}{(\partial x)}_{i+1/2,k} = \frac{2}{3} \sum_{j=1}^{3} \sum_{j=1/2,k} r_{j,1/2}^{i+1/2,k} C_{i+1/2,k-1/2}^{m_{i+1/2,k}}(j), n_{i+1/2,k}(j) \left( \frac{\partial C}{\partial x} \right)_{i+1/2,k} = \frac{2}{3} \sum_{j=1}^{3} \sum_{j=1/2,k} r_{j,1/2}^{i+1/2,k} C_{i+1/2,k-1/2}^{m_{i+1/2,k}}(j), n_{i+1/2,k}(j)
\]

\[
C_{i+1/2,k}^{m_{i+1/2,k}}(j), n_{i+1/2,k}(j) = C_{i+1/2,k}^{n} C_{i+1/2,k}^{n-1} C_{i+1/2,k}^{n-1} C_{i+1/2,k}^{n-1} C_{i+1/2,k}^{n-1} C_{i+1/2,k}^{n-1}
\]

\[
C_{i+1/2,k}^{m_{i+1/2,k}}(j), n_{i+1/2,k}(j) = C_{i+1/2,k}^{n} C_{i+1/2,k}^{n-1} C_{i+1/2,k}^{n-1} C_{i+1/2,k}^{n-1} C_{i+1/2,k}^{n-1} C_{i+1/2,k}^{n-1}
\]
where $D_k$ is the diffusion coefficient, $C_{i+1,k+1}^t$ and $C_{i+1,k}^t$ are the concentrations of virtual cells and $l_{j/k}$ are calculated based on Eq. (12). The diffusion flux is allocated to the neighboring cells depending on the common border length and the satisfying continuity, as mentioned for the advection term.

$$F_{i+\frac{1}{2},k} = \sum_{l=1}^{l_1} F_{i+\frac{1}{2},k} \Delta z_{i+\frac{1}{2},k,l} \quad l_i \geq 1, \quad \sum_{l=1}^{l_1} \Delta z_{i+\frac{1}{2},k,l} = \Delta z_{i+1,k}$$

where $F_{i+\frac{1}{2},k}$ is the diffusion flux at the control surface which is located at the block interface, and $F_{i+\frac{1}{2},k}$ is the flux distributed to its neighboring cells. $\Delta z_{i+\frac{1}{2},k}$ and $\Delta z_{i+\frac{1}{2},k,l}$ are the z-direction length of the control volume edge and the common border length to the neighboring cells at the block interface, respectively. The $F_{i+\frac{1}{2},k}$ amount is subtracted from the computational cell and the $F_{i+\frac{1}{2},k}$ amounts are added to the neighboring cells.

5. Model validation

5.1. Pure advection of a Gaussian distribution concentration

A two-dimensional bell-shape concentration distribution curve was assumed in a rectangular domain with the constant cross-section. The dimensions of the domain were $L(x,z)=(1440,900)$ m. The initial peak amount of concentration and its location were $C=1.0$ and $S(x,z)=(300,700)$ m, respectively. The velocity was taken to be $V(u,w)=(1.0,-0.5)$ m/s. The study domain was approximated using 48 cells and $dx=30$ m in the horizontal direction. Along the z-direction, 30 horizontal layers for the sigma scheme, and three blocks of 55, 30, and 15 horizontal layers for the multi-block sigma were used, as shown in Fig. 4, (a) and (b). Also, the test was solved by applying sigma-coordinate to compare the results of two different scheme of sigma and the multi-block sigma coordinates. The time step was $dt=20$s which met the Courant Number.

The initial distribution should be transported downstream into the channel without making any change in shape, and after time $T=900$, the peak should be located around $S(x,z)=(1200,250)$ m. The peak concentration is located at the right place in both sigma and multi-block sigma-coordinates, as shown in Fig. 4, (b) and (e). The 3D drawing of the bell-shaped concentration before and after advection in the multi-block sigma-coordinate scheme has been shown in Fig. 5, (a) and (b).

For assessing the results, statistical parameters including Average Relative Error (ARE), Root Mean Square Error (RMSE), and Maximum Relative Error (MRE) are used. The formulas for RE and RMSE are listed in Table 1.

The errors of the two schemes are listed in Table 2. The MRE of the multi-block sigma-coordinate was only about 9% more than that of the sigma-coordinate, considering the number of computational points decreased in the second and third blocks. The RMSE could not be determined in the multi-block sigma-coordinate due to the change in the number and location of the points in different blocks. The RMSE of the sigma-coordinate was about 0.08, which seems to be acceptable, and it is showing the accuracy of the sigma-coordinate. Therefore, the sigma-coordinate results were used as a reference to estimate the accuracy of the multi-block sigma-coordinate results. This analytical test showed that the multi-block sigma-coordinate can model the advection terms successfully.

5.2. Pure horizontal diffusion problem in a two dimensional $(x, z)$ sloping basin

The test was conducted for a pure horizontal diffusion problem in a two dimensional, closed, sloping basin. The basin was vertically stratified and initially static with no external forcing. The depths at the shallow and deep parts of the basin were 80 and 260 m, respectively. The length of the basin was $6000$ m; giving a bottom slope of 0.03. The vertical and horizontal diffusion coefficients were set equal to 0.0 and 5.0 m²/s, respectively. The initial salinity field was determined using the
The result of a pure advection test in a channel of the constant cross-section and slope for sigma coordinate and multi-block sigma coordinate. (a) and (b) are for sigma coordinate results and (c)–(e) are multi-block sigma coordinate.

The initial location of bell shape concentration is \( s=(300,700) \) in sigma coordinate.

The bell shape concentration is transformed to \( s=(750,475) \) after 450 s using velocity \( v=(1.0,-0.5) \) in sigma coordinate.

The bell shape concentration is located at \( s=(1200,250) \) after 900 s using velocity \( v=(1.0,-0.5) \) in multi-block sigma coordinate.

Following non-linear function.

\[
S(x, z) = S_{\text{max}} \left( \frac{h(x, z)}{H_{\text{max}}} \right)^{1/3}
\]

where \( h(x,z) \) is the water depth, \( H_{\text{max}} \) is the maximum depth of the basin, and \( S_{\text{max}} \) is the maximum salinity. The initial salinity field at the start of the simulation is shown in Fig. 6(a).

The initial horizontal salinity gradient was equal to zero; therefore, the salinity should not change with time. Any changes in salinity by solving the equation would have been due to the numerical errors in calculating the flux. This equation was
Fig. 6. The initial salinity field (a) prediction of salinity field (ppt) at the end of the 30-day simulation (f) and (c) and difference (ppt) between model-predicted salinity field at the end of the simulation and the initial salinity field (d) and (g) using sigma coordinate and multi-block sigma coordinate with their discretized domain (b) and (e).
solved for 30 days in both sigma and the multi-block sigma-coordinates. The study domain was approximated by using 12 × 500 m grids in the horizontal axes and seven equally divided layers in the vertical direction for the sigma scheme and three blocks of 5, 8, and 13 horizontal layers for the multi-block sigma-coordinate, as shown in Fig. 6 (b) and (e). The diffusion was set to zero along the horizontal wall boundaries. The results were presented in terms of the model predicted salinity field (ppt) at the end of the 30-day simulation, and the difference between the model predictions of salinity within 30 days and the initial field. The results of sigma–coordinate and multi-block sigma-coordinate are shown in Fig. 6 (c), (d), (f) and (g).

Two commonly used finite difference methods were applied to investigate the accuracy of the multi-block sigma-coordinate for solving diffusion: the linear interpolation [45], and the alternative transformation equations [46,11,12]. The results of the two methods are drawn in Fig. 7. Comparing the difference between the multi-block sigma-coordinate predictions of salinity (ppt) after 30 days and the initial field with the result of Mellor and Blumberg’s approximation [46], showed similar patterns of differences; but, the maximum difference of the multi-block sigma-coordinate was less than that of Mellor and Blumberg’s approximation, as listed in Table 3.

As shown in Fig. 6, the sigma-coordinate and the multi-block sigma-coordinate could have resulted in an underestimation of the salinity near the deeper depth and an overestimation at the shallower part. The MRE of the underestimation and overestimation are listed in Table 3.

The error ranges were very close in both methods, and the MRE of the sigma and the multi-block sigma-coordinates were 4.27 and 4.81 ppt, respectively, near the shallowest part of the basin. The MRE of sigma and the multi-block sigma-coordinates were about 30% less than the MRE of the linear interpolation [45] as well as Mellor and Blumberg’s approximation [46]. The maximum overestimation and underestimation relative errors of the multi-block sigma-coordinate were respectively more and less than the sigma–coordinate. The numerical error was due mainly to the inaccurate calculations
of the salinity horizontal gradient values. The results of the sigma-coordinate confirmed the multi-block sigma-coordinate accuracy for modeling the diffusion term. There was some disarray in the multi-block sigma-coordinate in comparison with the sigma-coordinate. The disarray, which makes sense, was caused by numerous efforts made to solve the diffusion flux and to calculate the horizontal gradient of the concentration from time $n$ at the block interfaces.

5.3. Laboratory trench experiment

The objective of this test was to verify the multi-block sigma-coordinate against the laboratory data of the trench experiment, as carried out by van Rijn (1987). At the upstream boundary, the water level and the mean velocity were 0.39 m and 0.51 m/s, respectively, and to maintain the equilibrium condition, sediment was supplied at a rate of 0.04 kg/sm. It was found that the rates of bed-load and equilibrium suspended sediment were $S_{b0}=0.01$ kg/sm and $S_{s0}=0.03$ kg/sm, respectively (Fig. 8, (a)). The sediment particles fall velocity and the characteristic diameter were measured in the laboratory and the value used in the simulation are $w_{s}=0.013$ m/s and $d_{50}=0.16$ mm, respectively. The effective bottom roughness was $k_s=0.025$ m, and the height of bedform was in a range of 0.015–0.035 m. The bed material porosity and density were equal to 0.4 and 2650 kg/m$^3$, respectively. The bed level change was measured after 15 h in the laboratory. At the inflow boundary, the equilibrium concentration profiles were described using the following equation,

$$\frac{C}{C_{be}} = \left( \frac{K_s}{z} \right) \frac{h-z}{K_s} \quad \text{for} \quad \frac{z}{h} < 0.5, \quad \frac{C}{C_{be}} = \left( \frac{K_s}{h-K_s} \right) \frac{z}{h-K_s} e^{-\frac{K_s}{h-K_s} z} \quad \text{for} \quad \frac{z}{h} \geq 0.5, \quad z_\ast = \frac{w_s}{0.4u_*}$$

(19)

More details of the model set-up were mentioned in van Rijn [47]. A logarithmic function represented the velocity profile at the inflow boundary. At the outflow boundary the “open boundary condition” was applied. The sigma-coordinate model was run using six horizontal layers, as shown in Fig. 8(b).

The multi-block sigma-coordinate assumed that if the water depth was less than 0.4 m, four horizontal layers will be applied; otherwise, the model would use eight horizontal layers (Fig. 11(a)).

$$\begin{cases} 
\text{if } h_{i,j} < 0.4 \text{ m } \Rightarrow 4 \text{ horizontal layers} \\
\text{if } h_{i,j} \geq 0.4 \text{ m } \Rightarrow 8 \text{ horizontal layers}
\end{cases}$$

(20)

Comparisons were made between the predicted sediment concentration profiles of the two models and the measured ones at four cross sections presented in Fig. 8(a), as shown in Fig. 9.

The longitudinal variation of the measured concentration profiles was simulated rather well by applying the two models. The measured concentration profiles agreed closely with the multi-block sigma profiles near bed zones, as well as with the sigma model profiles near the surface and middle depths.

Generally, the concentration profiles of the sigma model were matched with the measured concentration profiles more than the multi-block sigma-coordinate; however, considering the particular importance of the near bed concentration on deposition rate, it seems that the bed level of the multi-block sigma will match better with the measured bed level than the bed level simulated by the sigma coordinate after sedimentation. Fig. 10 shows the measured and computed bed levels for both models after 15 hr. The agreement between the measured and simulated bed levels was acceptable for both models; however, the multi-block sigma-coordinate profile was superior.

In the multi-block sigma-coordinate, modeling started with three blocks because of twice passing water depth from 0.4 m. By running the test, the positions of the block interface change by the morphological evolutions, till there were two
blocks at the end of the modeling process, whereas one block was omitted due to the trench erosion. The computation domains are shown in Fig. 11 after 6 h and 15 h, that show the multi-block sigma-coordinate can successfully follow the bed level evolutions and adapt the mesh.

To better investigate the accuracy and the efficiency of the multi-block sigma coordinate, a sigma–coordinate test was run using eight horizontal layers. The execution time, maximum overprediction and underprediction relative errors of the sigma and the multi-block sigma-coordinate and errors in estimating the “delta slope” and “delta position” are listed in Table 4. The delta slope referred to the angel of delta fore-set (the steep slope bed between the top-set and the bottom-set
of the delta) as described in Fig. 10(a). The error in simulating the delta position was represented by the error in the head of the delta bottom-set position. The delta slope and the position of the head of the bottom set in the experimental test were 33% and 11.5 m, respectively.

It can be seen that the multi-block sigma-coordinate decreased the computational cost without reducing the accuracy. The errors of the multi-block sigma-coordinate with 4-and-8-layers were better than those of the sigma-coordinate with six horizontal layers but the execution times were close. The maximum underprediction and overprediction relative errors of the multi-block sigma-coordinate were better than that of the 6-layer sigma-coordinate about 17% and 23%, respectively.
The delta slope and position were also about 27% and 2% better. The execution time was increased by almost 15% through using the multi-block sigma coordinate. The errors of the multi-block sigma-coordinate were close to that of the sigma-coordinate with eight layers, but its execution time was about 42% better. The maximum underprediction relative error of the multi-block sigma-coordinate was similar to that of the sigma-coordinate with eight layers, however, the maximum overprediction relative error was about 11% better. The delta position of the sigma-coordinate with eight layers was about 5% better than the multi-block sigma-coordinate, but the delta slope of the multi-block sigma-coordinate was about 15% better. The analysis showed that the multi-block sigma-coordinate had an accuracy almost equal to that of the sigma-coordinate with more horizontal layers, but the numerical cost was near the sigma-coordinate with less horizontal layers.

5.4. Hamidieh regulated reservoir

The multi-block sigma coordinate was applied for simulating a physical model of Hamidieh Regulated Reservoir. Hamidieh Reservoir is located on the Karkheh River, 11 km from Hamidieh Town, Khuzestan Province, Iran. Hamidieh Dam is 4.5 m high, 192 m long with 19 spillways, ten sluice gates and two water intakes which supply water for Azadegan and Chamran regions. Chamran water intake with an inlet width of 86.6 m supplies a maximum discharge capacity of 90 m$^3$/s for the 2.5 km long channel, while Azadegan water intake with an inlet width of 56 m operates with a maximum discharge capacity of 75 m$^3$/s for the 10.8 km long channel. Fig. 12 shows a plan of the reservoir and its intakes [50]. An undistorted 1/20 scale physical model of Hamidieh Reservoir was constructed in 2002 [51]. The model was aimed to investigate the operation of the system and to understand the sediment behavior. The bathymetry of Hamidieh Reservoir in the physical model is shown in Fig. 13(a). Table 5, lists the hydraulic and sediment information used to set-up the numerical model. Physical observations revealed that the dominant process of sediment was the suspended sediment transport and the majority of the sediment was non-cohesive [50]. For the sigma-coordinate, the water depth was divided into five layers, and the threshold height, which was equal to 0.15 m, was used to divide the depth into four and six layers for the multi-block sigma-coordinate. The grid size was selected as 0.25 m leading to 79 × 130 grid points in the plan. Fig. 13(c) shows a plan view of the number of
vertical layers in the multi-block sigma coordinate. The roughness parameter was used for calibrating the numerical results with regard to the physical tests \((k_s = 0.043 \text{ m})\), and the Chezy coefficient was determined from \(C = 18\log (12h/k_s)\).

The numerical modeling of the sediment transport process was assessed using the scenario no. 1 and predictions were compared to the physical model results at three sites as shown in Fig. 14(a). The pattern of the suspended sediment concentration near the bed is shown in Fig. 14(b). The concentration distribution throughout the cross-section cutting the location of the measured sites and the concentration distributed in depth direction are shown in Fig. 14 (c) and (d). Table 6 lists a comparison made between the measured and calculated suspended sediment concentrations in different points shown in Fig. 14(a) in the sigma and the multi-block sigma-coordinate. The Maximum Relative Errors of sigma and the multi-block sigma-coordinate were 19% and 23%, and Average Relative Errors were about 9% and 11%, respectively. The errors were in an acceptable level of agreement considering the complexity of the sedimentation process.

Scenario no. 2 which was 48.5 h, was used for modeling bed level evolutions. Fig. 15 shows the number of vertical layers (Fig. 15(a)) and the new bed level after scenario no. 2 in the multi-block sigma-coordinate (Fig. 15(b)).

---

**Table 5**
The hydraulic and sediment transport parameter used to set-up the numerical model for various scenarios in Hamidieh Reservoir scale model.

<table>
<thead>
<tr>
<th>Scenario no.</th>
<th>Boundary conditions</th>
<th>Field data</th>
<th>Physical model</th>
<th>Considerations</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Reservoir</td>
<td>Water discharge</td>
<td>156(m³/s)</td>
<td>87.2 (l/s)</td>
</tr>
<tr>
<td></td>
<td>Water level</td>
<td>20.45 (m)</td>
<td>1.0225 (m)</td>
<td>0.4724 (gr/l)</td>
</tr>
<tr>
<td></td>
<td>Concentration discharge</td>
<td>0.4724 (kg/ m³)</td>
<td>1.0225 (gr/l)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Discharge</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td></td>
<td>Azadegan Intake</td>
<td>Water level</td>
<td>20.00 (m)</td>
<td>1.0000 (m)</td>
</tr>
<tr>
<td></td>
<td>Water level</td>
<td>75 (m³/s)</td>
<td>41.9 (l/s)</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Reservoir</td>
<td>Water level</td>
<td>127(m³/s)</td>
<td>7.09 (l/s)</td>
</tr>
<tr>
<td></td>
<td>Water level</td>
<td>20.45 (m)</td>
<td>1.0225 (m)</td>
<td>0.2321 (gr/l)</td>
</tr>
<tr>
<td></td>
<td>Concentration discharge</td>
<td>0.2321 (kg/ m³)</td>
<td>1.0225 (gr/l)</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Discharge</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td></td>
<td>Azadegan Intake</td>
<td>Water level</td>
<td>20.45 (m)</td>
<td>1.0225 (m)</td>
</tr>
<tr>
<td></td>
<td>Water level</td>
<td>75 (m³/s)</td>
<td>41.9 (l/s)</td>
<td></td>
</tr>
<tr>
<td>Sediment particle</td>
<td>Water level</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
<tr>
<td>General data</td>
<td>Processor Properties: Intel® Core™i5 CPU M 480 @ 2.67 GHz</td>
<td>–</td>
<td>–</td>
<td>–</td>
</tr>
</tbody>
</table>

---

**Fig. 13.** The 2D bathymetry of Hamidieh Reservoir scale model (a), the computed flow pattern and velocity contours (b), and the number of vertical cells used in multi-block sigma coordinate at the initial condition (c).
Fig. 14. The location of measured sites at the physical model (a) [52]; the simulated patterns of the suspended sediments concentrations near the bed (b); the calculated suspended sediment concentration along the cross section and depth in sigma coordinate and multi-block sigma coordinate (c and d) after running Scenario no. 1 for the Hamidieh Reservoir scale model.

Fig. 15. The number of vertical layers new (a), the computed bed level elevation (b), and the calculated bed level aggradation (c) using the multi-block sigma coordinate after running Scenario no. 2 for the Hamidieh Reservoir scale model.
Table 6
A comparison between the measured and calculated suspended sediments concentrations in Hamidieh Reservoir scale model at the sites shown in Fig. 15, a for sigma and multi-block sigma coordinate.

<table>
<thead>
<tr>
<th>Point location in reservoir</th>
<th>Measured physical model (gr/l)</th>
<th>Sigma coordinate</th>
<th></th>
<th>Sigma coordinate</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>CRL</td>
<td>Calculated numerical model (gr/l)</td>
<td>Relative error (%)</td>
<td>CRL</td>
</tr>
<tr>
<td></td>
<td>0.410</td>
<td>0.424</td>
<td>3</td>
<td>0.430</td>
</tr>
<tr>
<td></td>
<td>CRM</td>
<td>0.423</td>
<td>19</td>
<td>0.556</td>
</tr>
<tr>
<td></td>
<td>CRR</td>
<td>0.417</td>
<td>4</td>
<td>0.437</td>
</tr>
</tbody>
</table>

![Profiles](image-url)  
Fig. 16. The comparison between the multi-block sigma coordinate predictions and the results of the physical model, sigma coordinate and Faghihirad [52] at 6 section profiles for the Hamidieh Reservoir scale model after running Scenario no. 2.

Model simulations were compared to the physical model resulting in six section profiles specified in Fig. 12(b) [51], and with the numerical results of Faghihirad et al. [52] at three sections which had the maximum bed evolutions. Faghihirad et al. [52] simulated the Hamidieh Reservoir scale model, using a depth-integrated scheme with the k-ε model for hydrodynamics, and a 3D layer-integrated model to predict the morphological processes. It can be seen that multi-block sigma-coordinate results were generally acceptable and close to the sigma coordinate and experimental data. At most points, the computational results of bed level changes were less than the physical results (Fig. 16). The MRE of sigma coordinate and the multi-block
sigma-coordinate as compared to the experimental data was about 30% at the left side of the L6-R6 profile, whereas, the MRE of multi-block sigma-coordinate as compared to the sigma-coordinate was about 4%. Fig. 17 shows the calculated suspended sediment concentration near the bed in the multi-block sigma-coordinate after 48.5 h of running scenario no. 2 (Fig. 17(a)), the number of vertical layers and the distribution of concentration in the depth direction at three cross-sections (Fig. 17(b–d)). A general investigation revealed that the multi-block sigma-coordinate observed all main features of the physical model with an acceptable range of accuracy and the generated error by applying the multi-block sigma coordinate can be ignored due to the inherent inaccuracies introduced by using empirical formulas and various assumptions in the sediment models.

6. Conclusions

The aim of this study was to introduce a multi-block sigma-coordinate scheme and to investigate its validation, accuracy, and application for solving the transport equation. The good agreement between the sigma and the multi-block sigma-coordinates in the four numerical tests showed that the multi-block sigma-coordinate can be an instrumental method and its accuracy was acceptable. The multi-block sigma-coordinates can be very effective for the long-term modeling of lakes and dam reservoirs whose bottom slopes are steep and the point aggregation in shallow parts increases the computational costs. This scheme can fit the topography properly and follow its evolutions, without producing the artificial flux at the steep bottom; it also decreases computational costs without reducing the accuracy. The multi-blocks sigma-coordinate achievement is due to the ability to change the size of the computational mesh according to the bed level evolution and the demanded accuracy. In shallow parts or the zones where less accuracy is needed, the mesh size can be fine. When there are morphological evolutions, the scheme can be adapted by moving the block interface or adding or omitting blocks for saving efficiency. Extending this scheme, for simulating the long-term reservoir sedimentation is the future study subject of the authors.

References


[27] A.C. Giapta, J.A. Slavicek, A. Bharti, A. Faith, for reducing pressure gradient errors improving the sigma coordinate stretching function: an idealized flow patterned after the Libyan near-shore region with the POM, Ocean Model. 23 (1–2) (2008) 59–72.


