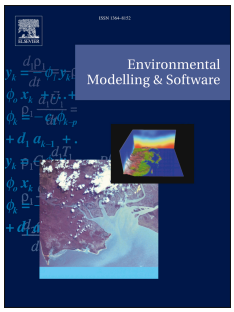
Journal Pre-proof



A MATLAB software platform for modelling vertically-integrated non-hydrostatic flows with moment equations

Pedro Gamero, Rafael J. Bergillos, Francisco N. Cantero-Chinchilla, Oscar Castro-Orgaz

PII: S1364-8152(19)30849-7

DOI: https://doi.org/10.1016/j.envsoft.2020.104674

Reference: ENSO 104674

To appear in: *Environmental Modelling and Software*

Received Date: 5 September 2019

Revised Date: 27 January 2020

Accepted Date: 21 February 2020

Please cite this article as: Gamero, P., Bergillos, R.J., Cantero-Chinchilla, F.N., Castro-Orgaz, O., A MATLAB software platform for modelling vertically-integrated non-hydrostatic flows with moment equations, *Environmental Modelling and Software* (2020), doi: https://doi.org/10.1016/j.envsoft.2020.104674.

A MATLAB software platform for modelling vertically-integrated non-hydrostatic flows with moment equations

Pedro Gamero[a], Rafael J. Bergillos[a]*, Francisco N. Cantero-Chinchilla[b], Oscar Castro-Orgaz[a]

[a] *Hydraulic Engineering Area, Department of Agronomy, University of Córdoba, Rabanales Campus, Leonardo da Vinci Building, 14071 Córdoba, Spain*

[b] *Soil and Water Division, IAS-CSIC, Spanish National Research Council, Alameda del Obispo s/n, 14004 Córdoba, Spain*

*Corresponding author.
E-mail address: rafael.bergillos@uco.es (R.J. Bergillos)

Preprint submitted to Environmental Modelling & Software

Abstract

This work presents a software platform to compute depth-integrated non-hydrostatic coastal and open channel flows. The software is based on the **V**ertically-**A**veraged and **M**oment (VAM) equations model. The VAM model uses the weighted residual method to account for the non-hydrostaticity and non-uniformity of flow. After presenting the governing equations, numerical scheme and structure of the graphical user interface, the software is applied to solve coastal and open channel flow tests with significant non-hydrostaticity and non-uniformity of flow. The results are compared with hydrostatic model computations and laboratory measurements. Where a hydrostatic-based simulation poorly reproduces experimental observations, the higher-order model equations by this software platform generate excellent predictions. These findings highlight the potential of this user-friendly software platform for modeling open channel, river and nearshore flows and processes, where non-hydrostatic pressure and non-uniform velocity shall be accounted for, which entails a meaningful advance for the hydraulic community.

Keywords: non-hydrostatic flows; depth-integrated model; software platform; coastal applications; open channel tests

Software availability

- Name of software: VAM Model Software
- Contact emails: ag2caoro@uco.es, fncantero@ias.csic.es, rafael.bergillos@uco.es, ag2gaojp@uco.es
- Requirements: MATLAB Runtime
- Availability: The software platform is freely available at https://github.com/Frncch/VAM_Model_Software

1. Introduction

Most of the user-friendly models used for the simulation of open channel flows are based on the solution of the Saint-Venant equations, where it is assumed that the distribution of fluid pressure is hydrostatic and the vertical distribution of the velocity component in the direction of motion is uniform (U. S. Corps of Engineers, 2002; Bladé et al., 2014; Teng et al., 2017). In free surface flows with a vertical length scale negligible as compared to the horizontal one, the Saint-Venant based models provide accurate approximations. However, in open channel transitions, near hydraulic structures, in dam break waves over movable beds, in eroded granular dikes by overtopping, or in river flows with a significant variation of bottom topography, among others, the variation of the vertical velocity is non-negligible and the vertical acceleration is, therefore, non-zero. Thus, the pressure distribution is non-hydrostatic with non-negligible dispersive effects in the equations of motion (Castro-Orgaz and Hager, 2017). The inclusion of such dispersive effects in a vertically-averaged framework allows to accurately solve a large portfolio of river and coastal flow processes (Kim and Lynett, 2011; Castro-Orgaz and Hager, 2017; Cantero-Chinchilla et al. 2018a; b). Coastal flows are typically simulated by using Boussinesq-type solvers, such as the short waves module in MIKE21 (Warren and Bach, 1992) or the Celeris model (Tavakkol and Lynett, 2017). The dam-break flood waves in rivers and the flow in the surf zone and the swash region are important

examples of non-hydrostatic flows, where Boussinesq equations have been extensively applied (Brocchini and Peregrine, 1996; Yamazaki et al., 2009; Cantero-Chinchilla et al., 2016). The use of non-hydrostatic models is required to provide realistic predictions in these flows (Wei and Jia, 2014a; b), albeit with a significant penalty in the numerical treatment of dispersive terms, due to the presence of higher-order derivatives of the depth-averaged velocity field (Castro-Orgaz and Hager 2017). However, given the still high computational cost of three-dimensional (non-hydrostatic) models for large-scale simulations, e.g. in a river basin, the development and use of depth-integrated simulation models, where the vertical coordinate is removed from the governing equations, is a common practice for non-hydrostatic modelling (Zijlema and Stelling, 2005; 2008; Yamazaki et al., 2009; Bai and Cheung, 2013; Wei and Jia, 2013; Aissiouene et al., 2015; Escalante et al., 2018, among others).

Essentially, three families of vertically-averaged models have been developed so far to simulate non-hydrostatic flows. The most widely used family is known as Boussinesq-type equations. This family considers the non-hydrostaticity through the inclusion of additional dispersive terms in the governing equations, demanding the numerical discretization of high-order velocity derivatives (Madsen et al., 1991; Madsen and Sørensen, 1992; Peregrine, 1966; Brocchini, 2013; Castro-Orgaz and Hager, 2017). The issue of increasing the order of spatial derivatives in a model appears subtle, but getting a robust solver when discretizing spatial derivatives of higher order than two is a formidable task when dealing with dam-break like numerical simulations. Another different (although related) family is the non-hydrostatic extension of the shallow water equations, which is based on a multilayer discretization of layer-averaged Reynolds-Averaged Navier-Stokes (RANS) equations (Stansby and Zhou, 1998; Casulli, 1999; Zijlema and Stelling, 2005; Fernandez-Nieto et al., 2018; Escalante et al., 2019). Although both Boussinesq equations and multilayer models are very useful tools, each one has its own disadvantages: the presence of high-order derivatives in the Boussinesq equations,

and the increased computational cost by increasing the number of layers to simulate a flow in the multilayer models. Finally, there is a third family of vertically-averaged models for the prediction of non-hydrostatic flows (Steffler and Jin, 1993; Khan and Steffler, 1996a; b), which is based on the use of the weighted residual method to derive the vertically-averaged and moment equations from the RANS equations. The modelling system of equations is known as **V**ertically **A**veraged and **M**oment (VAM) equations model (Cantero-Chinchilla et al., 2018a). The weighting function, $\Phi = 2(z-\overline{z})/h$, is used in the VAM model to derive the moment equations while applying the weighted residual method, where $\overline{z}$ is the elevation of the centroid of a section ($\overline{z} = z_b+h/2$). The VAM model is a physical system of equations that is still unfamiliar to the scientific community to some extent. It was developed by Steffler and Jin (1993) and then solved by Khan and Steffler (1996a; b) for flows in weirs and spillways, producing extremely accurate results for non-hydrostatic flows.

Apparently, publication of these results in hydraulics journals discouraged the use of VAM models by numerical modellers, who in this branch of science routinely apply Saint-Venant equations. The increased numbers of equations to solve in the VAM equations, and the apparent complexities in their structure when compared to Saint-Venant equations, unfairly relegated the use of VAM equations to a few isolated works conducted by their development team. However, complexities of a model are not in the number of equations to solve, but rather, on the mathematical structure of the terms to discretize. In this regard, the highest order derivative in the VAM equations is of order one, which defines a system of first order partial differential equations. This feature is attractive from the view point of numerical discretization, since higher order derivatives are absent. Discretization of dispersive terms is implicit for stability of the elliptic step of the solver, while the hyperbolic step is explicit and subject to the Courant-Friedrich-Lewy condition ($\mathrm{CFL} < 1$). Despite the VAM model was recently demonstrated to be extremely accurate predicting a number of river and coastal flows (Cantero-Chinchilla et al., 2018a); we acknowledge that the apparent complexities of the VAM model are a penalty in

terms of its diffusion and use by peers, since the development and application of the VAM equations and numerical scheme are time-consuming and not immediate tasks.

Therefore, the motivation of this paper is to present a user-friendly software platform that easily allows the application of the VAM model to simulate non-hydrostatic flows in a number of coastal and open channel flow cases. First, the governing equations and numerical scheme of the VAM model are shown in Section 2. Then, the structure of the graphical user interface (GUI) and the tests included in the software platform are described in Section 3. Section 4 shows several applications of the software platform. Finally, the conclusions of the work are summarized in Section 5. The VAM model software platform is freely available on GitHub (https://github.com/Frncch/VAM_Model_Software) and the MATLAB scripts of the VAM model are provided as supplementary material to this paper.

2. Theoretical and numerical basis of the VAM model

2.1 Governing equations

The governing equations of the one-dimensional (1D) VAM model are derived from the vertically-averaged RANS continuity equation, *x*- and *z*-momentum equations, and moment of *x*- and *z*-momentum equations (Steffler and Jin, 1993; Khan and Steffler, 1996a; b). They can be expressed in conservative form as follows (Cantero-Chinchilla *et al*., 2018a; see Hydraulic Reference Manual of the VAM model in the supplementary material for a complete derivation of the model equations):

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{F}}{\partial x} = \mathbf{S}_o + \mathbf{S}_\tau , \tag{1}$$

$$\mathbf{U}=\begin{bmatrix}U_1\\U_2\\U_3\\U_4\\U_5\end{bmatrix}=\begin{bmatrix}h\\q\\u_1\\h\bar{w}\\\dfrac{h^2 w_*}{12}\end{bmatrix},\mathbf{F}=\begin{bmatrix}q\\\dfrac{q^2}{h}+g\dfrac{h^2}{2}\\\dfrac{qu_1}{h}\\q\bar{w}\\\dfrac{hqw_*}{12}\end{bmatrix}=\begin{bmatrix}U_2\\\dfrac{U_2^2}{U_1}+g\dfrac{U_1^2}{2}\\U_2\dfrac{U_3}{U_1}\\U_2\dfrac{U_4}{U_1}\\U_2\dfrac{U_5}{U_1}\end{bmatrix}, \tag{2}$$

$$\mathbf{S}_o=\begin{bmatrix}0\\-\dfrac{\partial}{\partial x}\left(\dfrac{{u_1}^2 h}{3}+\dfrac{hp_1}{2\rho}+\dfrac{2hp_2}{3\rho}\right)-gh\dfrac{\partial z_b}{\partial x}-\dfrac{p_1}{\rho}\dfrac{\partial z_b}{\partial x}\\\dfrac{1}{2\rho}\left(\dfrac{\partial p_1}{\partial x}-\dfrac{p_1}{h}\dfrac{\partial h}{\partial x}\right)-\dfrac{4p_2}{h\rho}\dfrac{\partial \bar{z}}{\partial x}\\\dfrac{1}{6}\dfrac{\partial\left(hu_1 w_*\right)}{\partial x}+\dfrac{p_1}{\rho}\\\left[-\dfrac{h\bar{w}}{2}\dfrac{\partial q}{\partial x}+\left[q\bar{w}-\dfrac{hu_1 w_*}{6}\right]\dfrac{\partial \bar{z}}{\partial x}\right.\\\left.+\dfrac{\partial}{\partial x}\left[\dfrac{h^2 u_1}{10}\left(\bar{w}+\dfrac{w_b}{3}+\dfrac{w_s}{3}\right)\right]-h\overline{w^2}-\dfrac{2hp_2}{3\rho}\right]\end{bmatrix},\mathbf{S}_\tau=\begin{bmatrix}0\\-\dfrac{\tau_b}{\rho}\\0\\-\dfrac{\tau_b}{\rho}\dfrac{\partial z_b}{\partial x}+\dfrac{1}{2\rho}\dfrac{\partial h\tau_b}{\partial x}\\-\dfrac{h\tau_b}{2\rho}\left(\dfrac{\partial \bar{z}}{\partial x}+\dfrac{\partial z_b}{\partial x}\right)\end{bmatrix}, \tag{3}$$

where **U**, **F** and **S** are, respectively, the vector of unknowns, fluxes and source terms, (U_1,…,U_5) are the conservative flow variables, related to the conservation laws as follows: U_1 (continuity equation), U_2 (x-momentum equation), U_3 (moment of x-momentum equation), U_4 (z-momentum equation) and U_5 (moment of z-momentum equation). Subscripts o and τ refer to the inviscid and the turbulent stress source terms, respectively, x is the longitudinal coordinate, t is the time, ρ is the fluid density, g is the gravitational acceleration, q is the discharge per unit width, h is the flow depth, u_1 is the x-velocity at the free surface in excess of the mean q/h, z_b is the river bed elevation, $\bar{w}$ is the depth-averaged z-velocity [$\bar{w}$ = (w_s+w_b)/2 + 2w_2/3], with z-as the Cartesian vertical coordinate, w_s is the z-velocity at the free surface, w_b

is the z-velocity at the bed, w_2 is the mid-depth z-velocity in excess of the average between vertical velocities at the bed and free surface, w_* is the vertical velocity difference between bed and free surface ($w_* = w_b - w_s$), p_1 is the bed pressure in excess of the hydrostatic, p_2 is the mid-depth deviation from the linear non-hydrostatic law, τ_b is the bed shear stress, and $\overline{w^2} = (\bar{w})^2 + \frac{w_b^2}{12} + \frac{w_s^2}{12} - \frac{w_b w_s}{6} + \frac{1}{20}\left(2\bar{w} - w_b - w_s\right)^2$. Equations (1)–(3) are derived from the general vertically-integrated mass, momentum and moment of momentum equations, modelling the (x, z) velocity components and fluid pressure with distributions containing perturbation parameters $u_1(x, t)$, $w_2(x, t)$, $p_1(x, t)$ and $p_2(x, t)$ to deviate the flow from hydrostatic conditions with horizontal and depth-independent velocity. The particular choice used in Eqs. (1)–(3) is (Steffler and Jin 1993; Khan and Steffler 1996a; b; see Hydraulic Reference Manual of the VAM model in the supplementary material for a complete derivation of the model equations):

$$u(x,z,t) = u_0 + u_1\left(2\frac{z - z_b}{h} - 1\right), \tag{4}$$

$$w(x,z,t) = \left(w_b + w_2 4\frac{z - z_b}{h}\right)\left(1 - \frac{z - z_b}{h}\right) + w_s\frac{z - z_b}{h}, \tag{5}$$

$$p(x,z,t) = \left(\rho g h + p_1 + p_2 4\frac{z - z_b}{h}\right)\left(1 - \frac{z - z_b}{h}\right). \tag{6}$$

Note that the perturbation parameters are functions of x and t, but not of the vertical coordinate z. To mathematically close the model, additional physical statements are required. The kinematic boundary conditions at the bed and free surface levels are, respectively,

$$w_b = u_b\frac{\partial z_b}{\partial x} = \left(\frac{q}{h} - u_1\right)\frac{\partial z_b}{\partial x}, \tag{7}$$

$$w_s = \frac{\partial h}{\partial t} + u_s \frac{\partial z_s}{\partial x} = -\frac{\partial q}{\partial x} + \left(\frac{q}{h} + u_1 \right) \frac{\partial (z_b + h)}{\partial x}, \tag{8}$$

where u_b and u_s are the u-velocity components at the bed and free surface, respectively. Two reactive equations, i.e. where temporal derivates of conserved variable are absent, are prescribed to provide closure to the system of Eqs. (1)−(3). The first is obtained combining Eqs. (7)–(8), resulting

$$\frac{h^2 w_*}{12} = \frac{h^2}{12}\left(\frac{q}{h} - u_1 \right) \frac{\partial z_b}{\partial x} + \frac{h^2}{12} \frac{\partial q}{\partial x} - \frac{h^2}{12}\left(\frac{q}{h} + u_1 \right) \frac{\partial (z_b + h)}{\partial x}. \tag{9}$$

The second is the moment of continuity equation, which reads (see Hydraulic Reference Manual of the VAM model in the supplementary material):

$$h\overline{w} = -\frac{h}{2} \frac{\partial q}{\partial x} + q \frac{\partial \overline{z}}{\partial x} + \frac{1}{6} \frac{\partial (h^2 u_1)}{\partial x}. \tag{10}$$

The VAM model system (Eqs. 1−3 and Eqs. 9–10) is based on a simple turbulence closure where the normal stresses are neglected and the shear stress is considered to vary linearly between τ_b at bed and zero at the surface. The latter is modelled using the Manning's equation parametrized with the near-bed non-hydrostatic velocity field (Cantero-Chinchilla *et al*. 2018a; see also Hydraulic Reference Manual of the VAM model in the supplementary material). If $u_1 = w_2 = p_1 = p_2 = 0$ the VAM Eqs. (1)–(3) reduce to the Saint-Venant equations. In this case the moment equations and the vertical momentum equation are not needed, given the lack of perturbation parameters, e.g. only h and q are computed.

A significant feature of the VAM equations as compared to Boussinesq equations is that a wave breaking criteria is not needed, given that the vertical profile of horizontal velocity u is modeled, thereby introducing dissipation into the solution. The VAM model was demonstrated to produce the transition from solitary wave to

broken waves during run-up on a beach without any empirical formulation for wave breaking (Cantero-Chinchilla *et al*. 2018a). If $u_1 = w_2 = p_2 = 0$, the only perturbation parameter retained is p_1, and the VAM Eqs. (1)–(3) reduce to the Vertically-Averaged equations (VA) by removing the moment equations (Khan and Steffler 1996a). The VA system of equations is basically equivalent to a Boussinesq-type system, albeit with a factor affecting the dispersive term (Cantero-Chinchilla *et al*. 2016). The VAM equations can model with great accuracy dispersive effects up to kH = 8, where k is the wave number and H is the still water depth. However, the accuracy of the VA model is acceptable only to kH = 2 (Steffler and Jin, 1993; see Hydraulic Reference Manual of the VAM model in the supplementary material for a detailed comparison of VAM and VA models). Therefore, the advantage of the VAM model over other types of models, such as those based on Boussinesq equations, is that dispersive effects are modelled with great accuracy up to large kH values, and that wave breaking is automatically accounted for by the model equations. This comparison is not aimed at discouraging the use of Boussinesq-type equations, which are valuable numerical models by improving their linear dispersion relation and implementing a wave breaking criterion. The comparison focuses on presenting the VAM equations as a viable alternative to model the same flows, showing some of its important features.

2.2 Numerical scheme

The system of Eqs. (1)−(7) is solved using a semi-implicit hybrid finite volume-finite difference numerical scheme (Cantero-Chinchilla *et al*., 2018a; see supplementary material, where the MATLAB scripts are freely available. Fundamentals of the numerical method implemented are explained in detail the Hydraulic Reference Manual accompanying the VAM code). The solution strategy uses a splitting approach. The homogeneous part of Eq. (1), i.e. with no source terms, is solved first in an advection step using a Godunov-type finite-volume scheme. The

209 fourth-order total variation diminishing monotone upstream centered scheme for
210 conservation laws (MUSCL-TVD-4th) and the *minmod* limiter are used to reconstruct
211 the flow variables at the cell-interfaces (Yamamoto and Daiguji, 1993; Kim et al.,
212 2009; Kim and Lynett, 2011; Cantero-Chinchilla et al., 2016).

213 The weighted surface-depth gradient method (WSDGM) (Aureli et al., 2008) and a
214 discretization of the gravity term in the *x*-momentum equation to grant the C-
215 property of the numerical scheme are implemented. The numerical flux at the cell-
216 interface is determined using the HLLC approximate Riemann solver (Toro, 2001,
217 2009). In this step, the computed flow depth is the final value for the actual time. The
218 homogeneous solution is then updated by including the effects of the inviscid terms
219 using a centred implicit finite-difference scheme resulting from the application of the
220 backward Euler formula. The system of algebraic equations is iteratively solved
221 using the Newton-Raphson method. The equations resulting from this step are non-
222 linear and implicit, and, for this reason, an iterative solver is needed. The discretized
223 system of equations and Jacobian matrix implemented are described in the Hydraulic
224 Reference Manual accompanying the VAM MATLAB scripts as supplementary
225 material. Note that, in this step, the two reactive equations [Eqs. (9) and (10)] are
226 used for model closure. In a final step, the inviscid non-hydrostatic solution is
227 updated by including the effect of the turbulent source terms using a centred finite-
228 difference scheme and the forward Euler formula. The time step is subjected to the
229 CFL condition to ensure numerical stability, which was found by numerical
230 experimentation to be restricted to CFL< 0.5. A detailed description of the numerical
231 scheme is available in Cantero-Chinchilla et al. (2018a), and the MATLAB scripts
232 can be downloaded as supplementary material.

233 ## 3. Description and structure of the software platform

234 The initial window of the software platform is shown in Fig. 1. It allows selecting
235 coastal or open channel flow applications. Within the first option, two kinds of

problems can be modelled: (i) solitary wave propagation and (ii) solitary wave propagation and run-up on inclined planes (Fig. 2). On the other hand, the open channel flow applications consist of: (i) Favre waves and (ii) flow over Gaussian obstacles (Fig. 3). A flow chat with the inputs and outputs of each application of the software platform is shown in Fig. 4. These applications are further detailed in Sections 3.1 and 3.2.

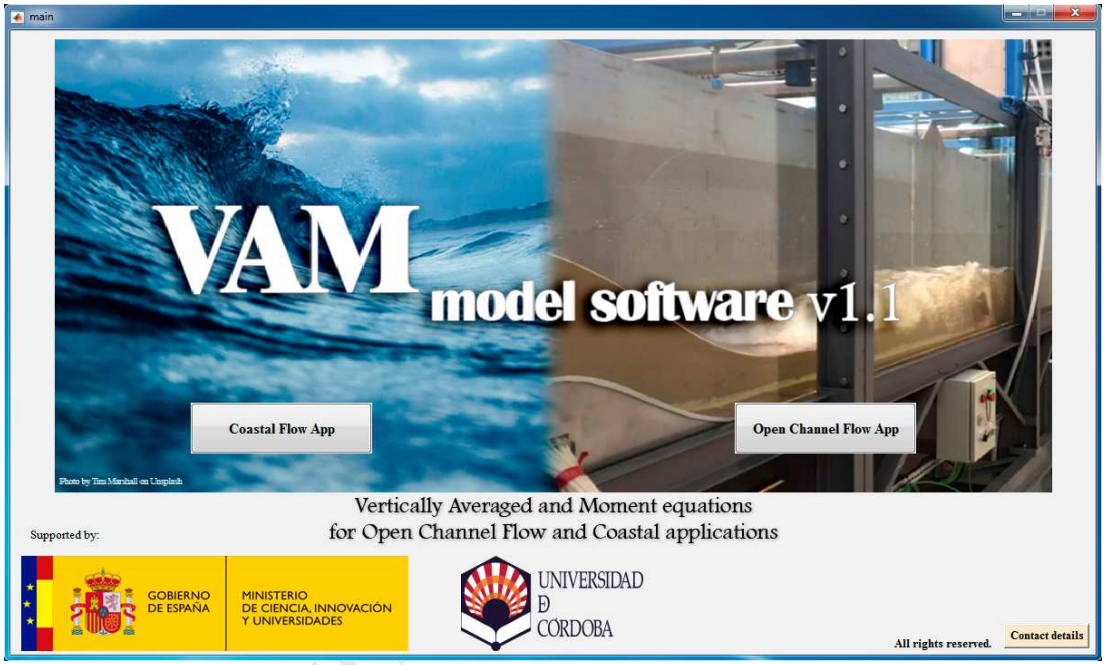
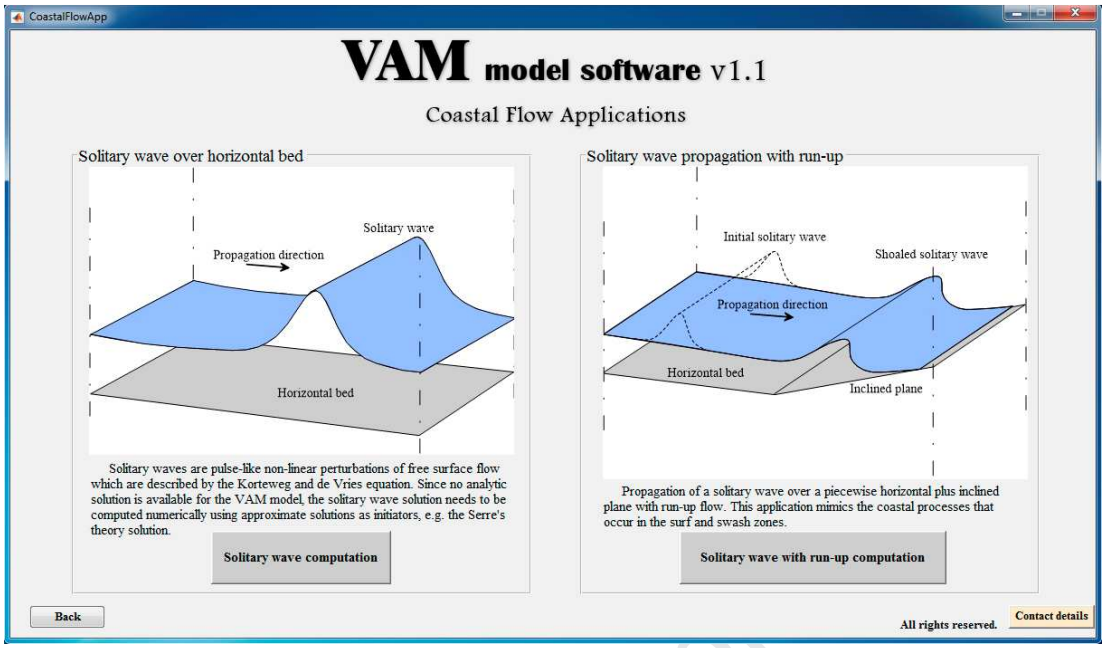
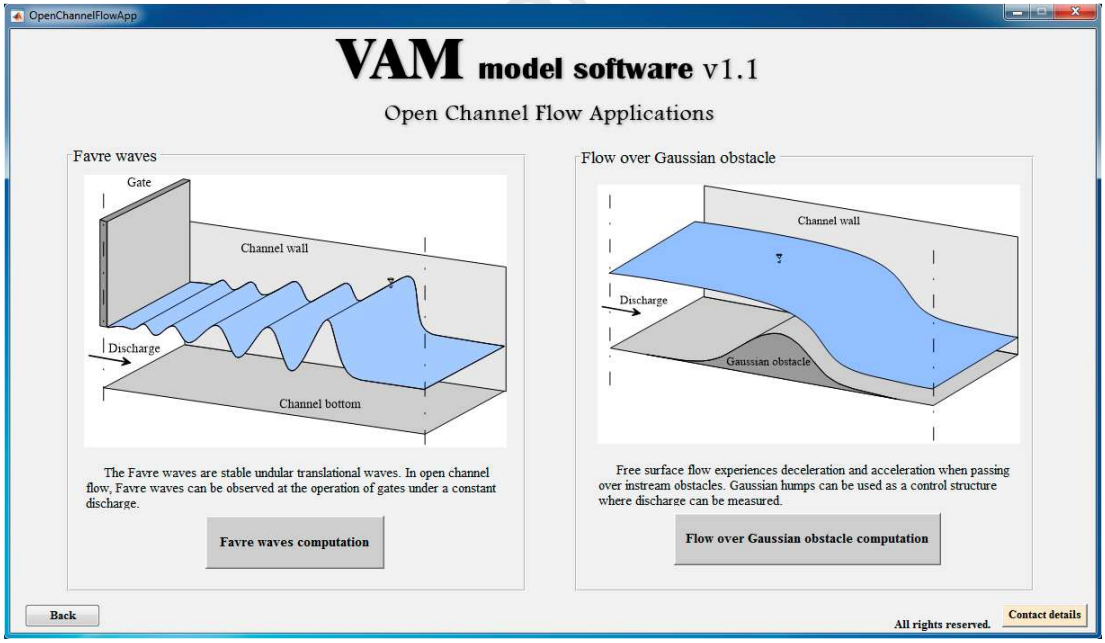


Figure 1. Initial window of the software platform.



244

245 **Figure 2**. Coastal flow applications window.



246

247 **Figure 3**. Open channel flow applications window.

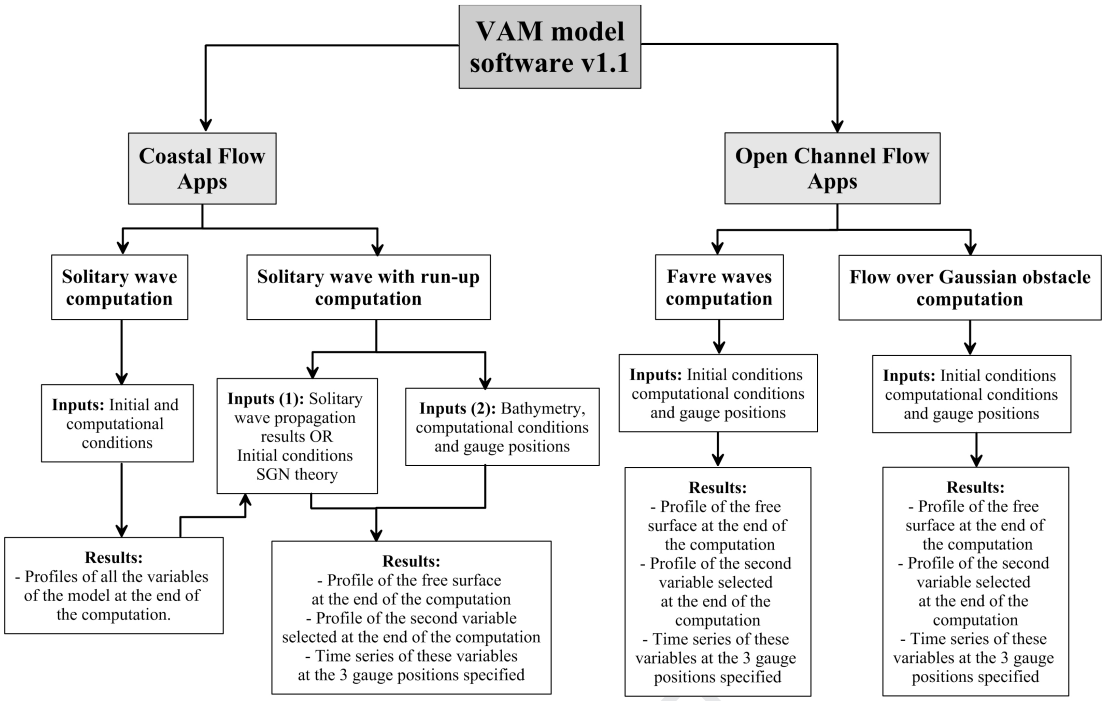


Figure 4. Flow chart with the input conditions and the results of each application.

3.1 Coastal flows

3.1.1 Solitary wave propagation

The generation and propagation of a solitary wave is a challenging test for numerical models, since a balance is required between non-linearity and dispersion to produce a wave propagating with a permanent form. Since there is not known analytical solution for the solitary wave solutions embedded into the VAM model, it is determined numerically as follows. The analytical solution of the Serre-Green-Naghdi equations (Castro-Orgaz and Hager, 2017) is considered as initial condition for the application of the VAM model. When the model is run with this initial condition, an unsteady motion involving a re-shaping of the wave profile occurs over a few time steps. After routing a few seconds a wave of stable form results, which is the solitary wave solution of the VAM model (Cantero-Chinchilla *et al*., 2018a).

The sketch of the problem is shown in Fig. 5. The following input variables are required: water depth (h), wave amplitude (A), length of the simulation reach (L), Courant-Friedrich-Lewy number (CFL), cell width (Δx) and simulation time (T_{max}). The user can select the second variable to monitor (q, u_0, u_1, p_1, p_2 or w_2) and the Y-limits of the graphs, as well as plot the initial conditions, stop the simulation and save the solitary wave solution to a file (Fig. 5).

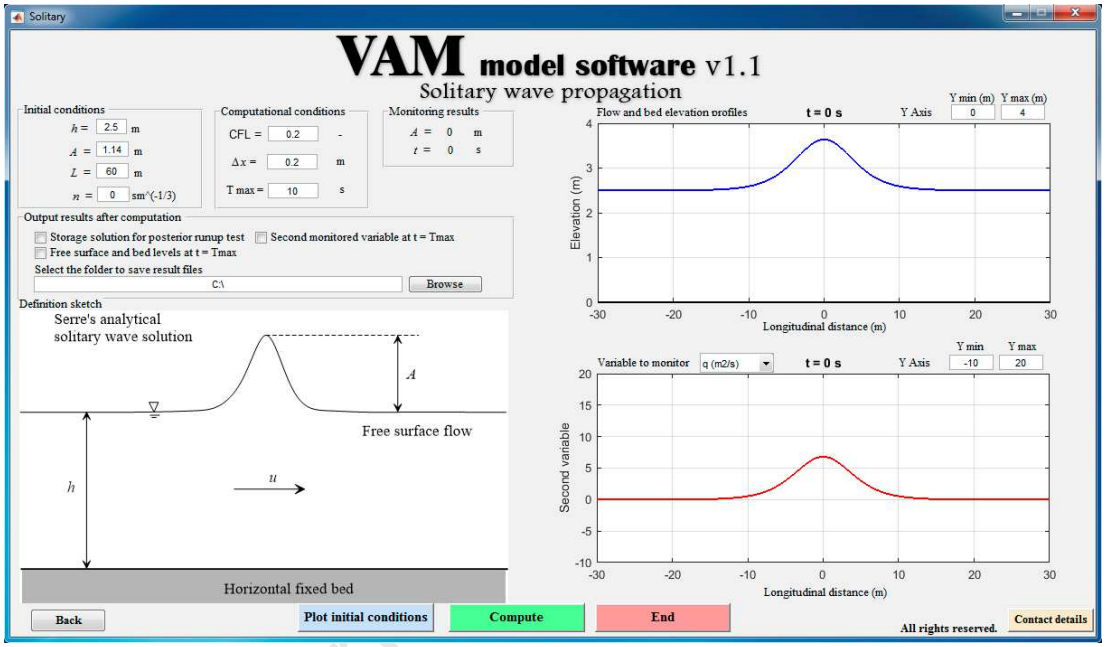


Figure 5. Window for computation of solitary wave propagation.

3.1.2 Solitary wave propagation and run-up on a plane beach

The second application of the software platform is the solitary wave propagation and run-up on a constant slope beach. This case is relevant since wave breaking and run-up flows are responsible of coastal erosion and flooding issues. The VAM model reproduces the wave breaking without using any empirical breaking criteria to switch off dispersive terms, as typically done in Boussinesq-type models (Brocchini 2013; Castro-Orgaz and Hager 2017). The wave breaking mimicking ability of the VAM model is due to the modeled velocity profile, which introduces differential advection

of momentum $[\frac{1}{3}\partial(u_1^2 h)/\partial x]$ and can, thus, counterbalance the effects of the dynamic component of fluid pressure, accounted for in the VAM model by the perturbation parameters p_1 and p_2. The geometrical configuration of the problem is shown in the lower-left part of Fig. 6. The initial conditions for this test can be imported from the solution of the solitary wave propagation problem detailed in Section 3.1.1. Alternatively, the analytical solution of the Serre-Green-Naghdi equations can be considered as initial condition through the definition of water depth (h), wave amplitude (A) and wave position (x_c), as can be observed in Fig. 6.

Apart from these inputs, the following variables should be specified in the interface: length of the horizontal stretch of the profile (L_h), length of the sloped stretch (L_s), Manning roughness coefficient (n), beach slope (S_0), Courant-Friedrich-Lewy number (CFL), cell width (Δx), simulation time (T_{max}) and positions of the three gauges indicated in the definition sketch (Lg_1, Lg_2 and Lg_3). The user can also plot the initial conditions, start and finish the simulation, plot the evolution of the free surface and the second variable previously selected, and save the results (Fig. 6).

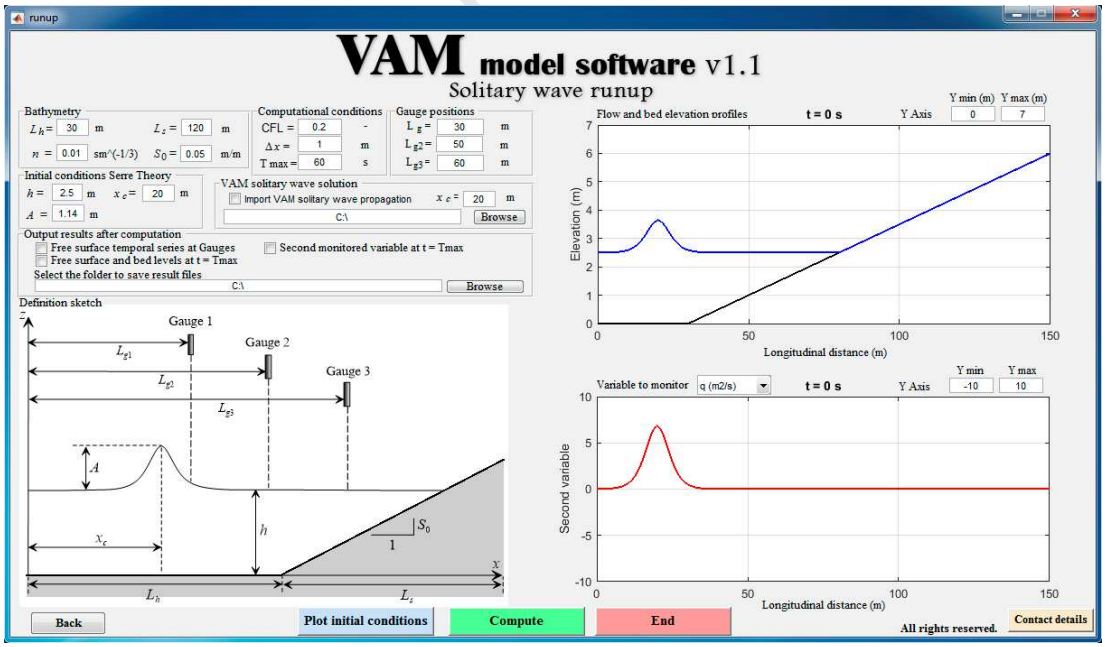


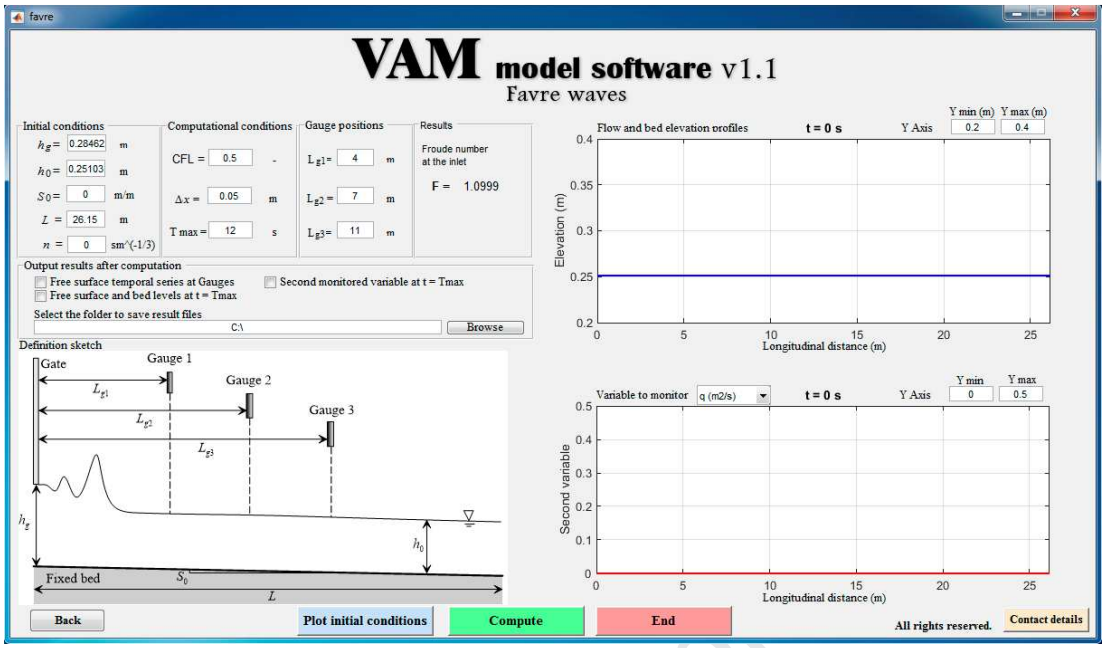
Figure 6. Window for computation of solitary wave run-up.

295 ## *3.2 Open channel flows*

296 ### *3.2.1 Favre waves*

297 Favre waves are dispersive trains of free-surface undulations generated at the head
298 of a bore or after a rapid opening/closing of a gate (Favre, 1935; Soares-Frazao and
299 Zech, 2002; Soares-Frazao and Guinot, 2008). Hydrostatic models are dispersionless
300 and, thus, unable to predict Favre waves. Therefore, Favre waves can only be
301 modelled in a depth-integrated framework by shallow-water models accounting for
302 non-hydrostatic pressures and, thus, dispersive effects (Soares-Frazao and Zech,
303 2002).

304 The first open channel flow application of the software platform allows simulating
305 Favre waves generated by the opening of a sluice gate. As can be observed in Fig. 7,
306 the input variables required to run this application are the distance from the bed to
307 the gate (h_g), downstream water depth (h_0), bed slope (S_0), length of the canal (L),
308 Manning roughness coefficient (n), Courant-Friedrich-Lewy number (CFL), cell
309 width (Δx), maximum time of the simulation (T_{max}) and positions of the gauges (Lg_1,
310 Lg_2, and Lg_3). The definition sketch of the lower-left part in Fig. 7 indicates the
311 geometrical configuration of the test, including some of the aforementioned
312 variables.



313

314 **Figure 7**. Window for computation of Favre waves.

315 The interface allows plotting the initial conditions as well as the evolution of the
316 free surface elevation and the second variable selected by the user during the
317 simulation period. The simulation is run by the user though the *Compute* button and
318 can be ended at any time by pressing the *End* button. The free surface temporal series
319 at the gauge positions as well as the spatial distribution of the free surface and the
320 second selected variable can be saved through the selection of the corresponding
321 items and the specification of the path to save the result files (Fig. 7).

322 *3.2.2 Flow over a Gaussian obstacle*

323 The last application allows reproducing the water waves evolving over a Gaussian
324 obstacle. The initial conditions are of dam-break type, with a significant variation in
325 the river depth downstream, as shown in the definition sketch of Fig. 8. This test
326 aims at mimicking the wave propagation over an obstacle, like those occurring over a
327 river bed mound when an undular bore propagates in a river. This type of water wave
328 problem was previously used by Nadiga et al. (1996) to test Boussinesq-type models,

given that it is challenging. The particular case of flow over obstacle with an initially uniform distribution of water depths can be also simulated with the software platform. For that, the two initial values of water depth specified by the user (denoted by h_u and h_d, see Fig. 8) should be equal.

Apart from these two values, the following input variables should be specified: constants for the mathematical definition of the Gaussian bed profile (a, b and x_c), Manning roughness coefficient (n), length of the canal (L), river flow per unit width (q), Courant-Friedrich-Lewy number (CFL), cell width (Δx), simulation time ($T_{\max}$) and gauge positions (Lg_1, Lg_2 and Lg_3).

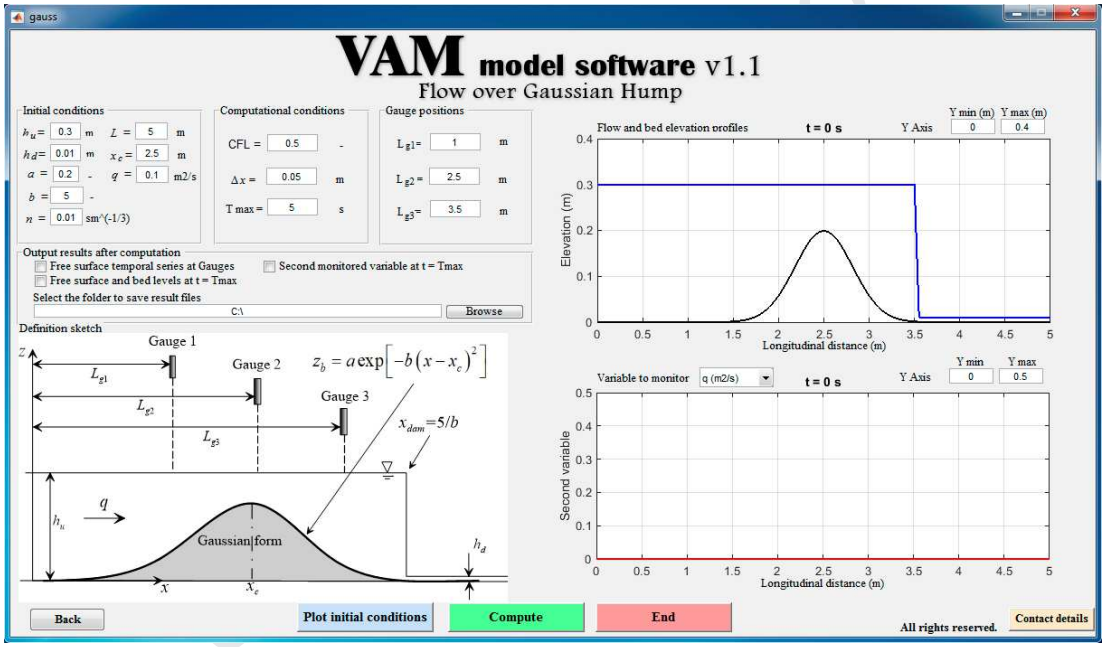


Figure 8. Window for computation of flows over Gaussian humps.

As in the rest of applications, the user can also depict the initial conditions through the *Plot initial conditions* button, start the simulation by means of the *Compute* button and end the computation by pressing the *End* button at any time. The evolution of the free surface elevation and the second variable selected by the user are depicted in the right part of the window, and the results can be also stored through the selection of the items and the specification of the path (Fig. 8).

The computational time of the software platform depends on the computational conditions specified for each case, that is, on the Courant-Friedrich-Lewy number (CFL), the number of cells used, and the maximum simulation time (T_{max}). The computational time is also dependent on the characteristics of the computer employed for the simulations.

4. Application examples of the software platform

4.1 Solitary wave propagation

This section reports an application of the *solitary wave propagation* module and compares the results with the hydrostatic version of the VAM model. The initial conditions settled in the software platform window (Fig. 5) were h = 2.5 m, A = 1.14 m, L = 80 m, n = 0 $sm^{-1/3}$, CFL = 0.1 and T_{max} = 10 s. Fig. 9 shows the initial wave and the solitary wave solutions with the hydrostatic and non-hydrostatic versions of the VAM model. It is observed that, after 25 seconds, the wave propagated over the horizontal fixed bed is transformed into a shock by the hydrostatic computation, whereas the non-hydrostatic wave propagation solution obtained with the VAM software platform produces a solitary wave. The train of small ripples generated at the tail of the solitary wave is produced during the transformation of the initial solitary wave (analytical solution of the Serre-Green-Naghdi equations) into the solitary wave solution of the VAM equations. If a large computational time is prescribed, the ripples are splitted from the solitary wave tail, and the solitary wave continues its propagation isolated. This non-hydrostatic solution obtained by using the *solitary wave propagation* module can be imported to *the solitary wave with run-up* module, as it is detailed in Section 4.2.

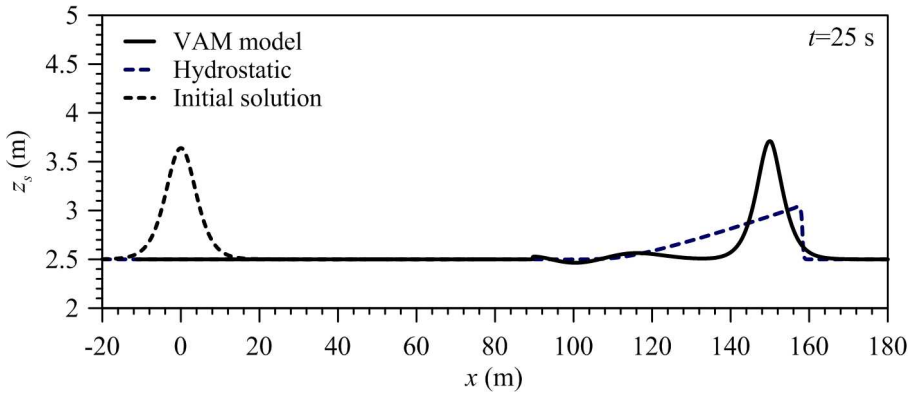


Figure 9. Solitary wave propagation obtained with the hydrostatic (grey solid line) and non-hydrostatic (black solid line) versions of the VAM model after 25 s of routing. The initial solution is also shown (black dashed line).

4.2 Solitary wave run-up on plane beach

This section deals with the application of the present software platform to address solitary wave run-up on a plane beach. The experimental data of solitary wave run-up by Synolakis (1986) are selected to evaluate the accuracy of the corresponding module of the software platform. The experiments were conducted in a wave tank of the W. M. Keck Laboratories at the California Institute of Technology. The dimensions of the wave tank were 37.73 m (length) x 0.61 m (height) x 0.39 m (width). At a distance of 14.68 m from the wave generator, an inclined plane beach of slope 1:19.85 was installed, see Fig. 10.

From the experimental dataset collected by Synolakis (1986), a solitary wave of A/h = 0.3 with h = 1 m was selected here. This wave amplitude will introduce significant non-linear effects on the solitary wave propagation. First, the solitary wave is propagated over a horizontal frictionless bed using the *solitary wave propagation* module of the software platform. This module uses a moving-grid subroutine that allows the propagation of the solitary wave in a short length, which is usually defined to produce an unaltered non-hydrostatic variable p_1 at the moving

boundaries. As the solitary wave amplitude is continuously monitored in the screen, the program can be stopped when the user identifies the desired amplitude and, then, the output result file is automatically generated. Later, the output of the *solitary wave propagation* module is used as initial condition in the *solitary wave propagation with run-up* module. This module was run six times to compare the results with the data by Synolakis (1986), setting the maximum computational time at t = 4.7891 s (Fig. 10a), t = 6.3855 s (Fig. 10b), t = 7.9819 s (Fig. 10c), t = 15.963 s (Fig. 10d), t = 17.561 s (Fig. 10e), and t = 19.156 s (Fig. 10f).

As observed by Synolakis (1986), during the propagation of the solitary wave along the plane beach, the wave deformed and the skewness increased, leading to an instability that induced the wave breaking. The software results are in good agreement with laboratory measurements, resulting, however, a minor phase shift at the hydraulic jump position formed during the drawdown phase at t = 17.561 s (Fig. 10e). Note that when a hydraulic jump is formed, the turbulence is significant and, thus, the simple turbulence module implemented in the VAM model may not be accurate enough. Further, the present vertically-averaged computational results cannot reproduce the inherent three-dimensional flow effects in hydraulics jumps. Therefore, despite the limitations, the computational results obtained with the VAM model are considered satisfactory in a depth-integrated sense. As in the case of the solitary wave propagation (Section 4.1), the agreements between the software platform results and the experimental data are significantly more satisfactory than those obtained with the hydrostatic version of the VAM model, especially before the inception of wave breaking (Fig. 10).

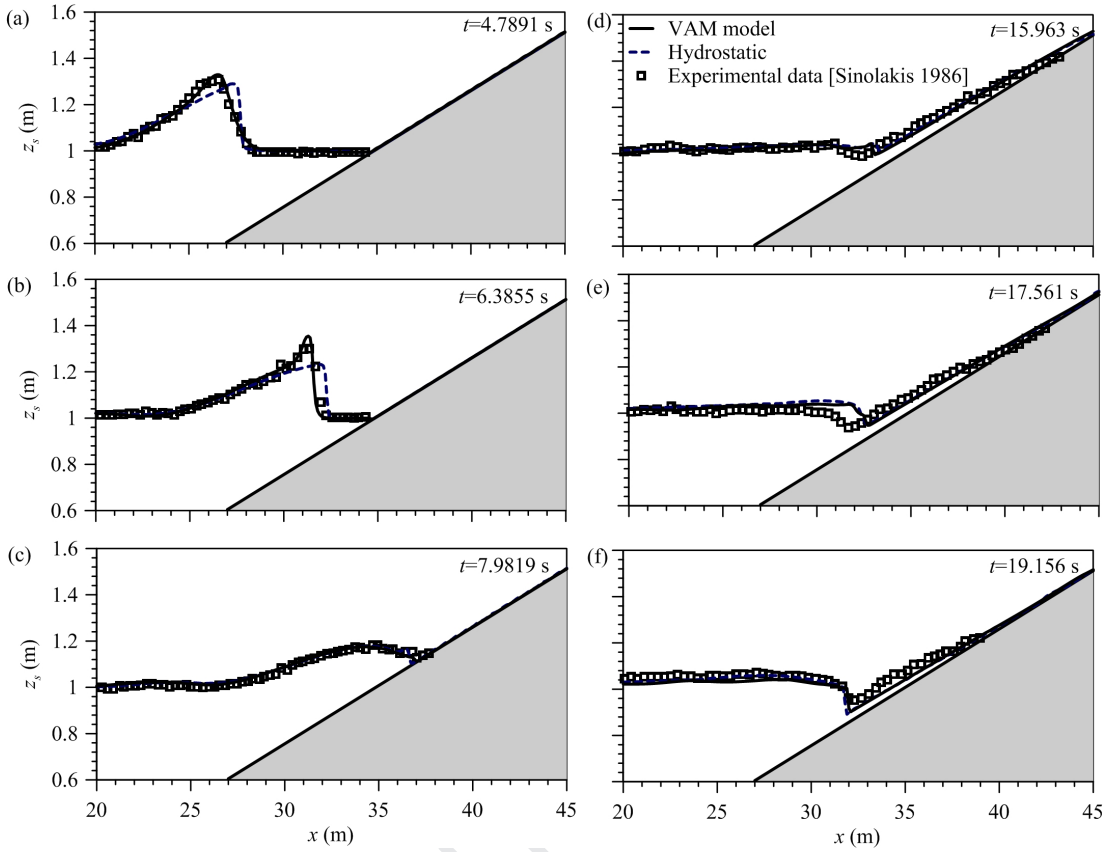


Figure 10. Solitary wave run-up obtained with the hydrostatic (grey dashed line) and non-hydrostatic (black solid line) versions of the VAM model at different times. Comparisons with experimental measurements by Synolakis (1986) (black squares).

4.3 Favre waves

In this section, the software platform is tested to estimate three different data sets of Favre waves generated by using different Froude numbers for the inlet conditions: F = 1.081 (Fig. 11a), F = 1.104 (Fig. 11b), and F = 1.192 (Fig. 11c). Once the flow depth at rest in the channel is set (h_0), the inlet tailgate depth (h_g) is calculated using the expression by Soares-Frazão and Zech (2002), i.e. $F = [h_g\,(h_g + h_0) / (2h_0^2)]^{1/2}$. The input variables for the software platform in the *Favre waves* module to generate Fig. 11 are shown in Table 1.

The non-hydrostatic free-surface predictions of the VAM model obtained with the software platform are compared with the experimental data by Soares-Frazão and Zech (2002). These data were collected in an experimental flume of the laboratory of the Civil Engineering Department at the University of Louvain. A gate located at a 10 m distance from the beginning of the flume separates two regions: the upstream reservoir and the channel of length 26.15 m. The gate was suddenly opened and the free surface measured at different downstream locations. Measurements at distances from the gate of 13.15 m, 15.65 m and 13.5 m were used for the tests with F = 1.081, F = 1.104 and F = 1.192, respectively (Table 1).

To extend the comparison, the hydrostatic computation accomplished using the main code of the VAM model, i.e. neglecting p_1, p_2, u_1 and w_2, and thus equivalent to solving Saint-Venant equations, is also depicted in Fig. 11. In addition, the results obtained by applying the widely-used (hydrostatic) IBER model software for river flows (Bladé et al., 2014; Cea and Bladé, 2015; Cea et al., 2016) are included to allow the readers to compare the results between both hydraulics software platforms. The results by IBER were generated using the input parameters indicated in Table 1. The 2D geometry in the IBER model was set to keep invariant the transverse direction (i.e. resembling 1D conditions). The comparisons between the time series of the free surface obtained for the three Froude numbers are shown in Fig. 11.

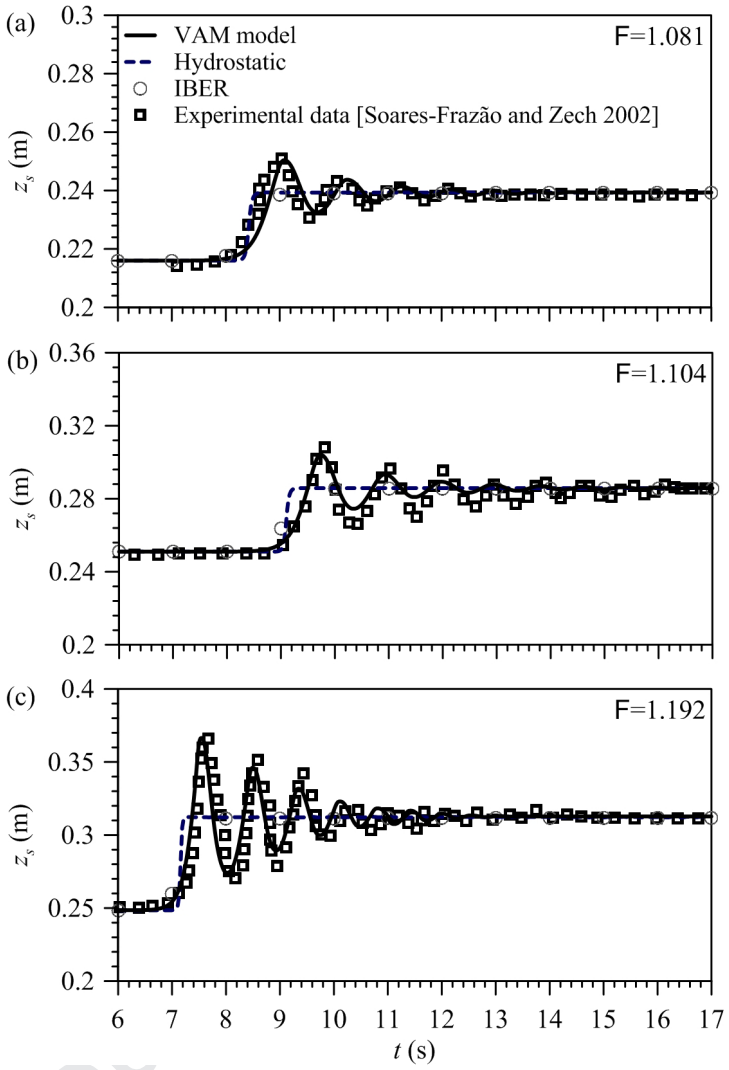
Table 1. Input variables for the *Favre* waves module of the software platform to generate Fig. 11.

F	**1.081**	**1.104**	**1.192**
h_g (m)	0.2394	0.286	0.3127
h_0 (m)	0.216	0.251	0.2485
S_0	0	0	0
L (m)	40	40	40

n (s/m$^{1/3}$)	0.01	0.01	0.01
CFL	0.2	0.2	0.2
Δx (m)	0.05	0.05	0.05
T_{max} (s)	17	17	17
L_{g1} (m)	13.15	15.65	13.15

445

446 In Fig. 11 it is observed that the results of the VAM model software platform
447 produce excellent approximations to the laboratory measurements by Soares-Frazão
448 and Zech (2002) in the three cases for the whole time series. On the contrary, the
449 agreements between experimental data and results obtained with both the hydrostatic
450 version of the VAM model and the IBER model are significantly worse, due to the
451 lack of non-hydrostatic modelling. Thus, Favre waves are not captured by hydrostatic
452 simulations. This remarks the importance and necessity of using non-hydrostatic
453 models, as the VAM model software platform presented in this work, to predict
454 Favre wave during gate manoeuvres in open channel flows.



455

456 **Figure 11**. Depth-hydrographs of Favre waves obtained with the hydrostatic (grey
457 dashed line) and non-hydrostatic (black solid line) versions of the VAM model and
458 with the IBER model (grey dots). Comparisons with experimental measurements by
459 Soares-Frazão and Zech (2002) (black squares).

460 *4.4 Flows over a Gaussian obstacle*

461 This section addresses the flow over Gaussian obstacle for three different values of
462 the discharge per unit width (q = 0.0561 m^2/s, q = 0.07703 m^2/s and q=0.10144 m^2/s)

experimentally tested by Sivakumaran (1981). The experiments were conducted in a laboratory flume with dimensions of 9.15 m (length) x 0.75 m (height) x 0.44 m (width). A curved bed with a maximum height of 0.2 m was built in the flume. Several values of unit discharge were tested, and the free surface position and bed pressure head were measured for each run at different positions along the flume.

The input variables for the software platform in the Flow over Gaussian obstacle module are shown in Table 2. Fig. 12 depicts the comparison between the experimental data of the free surface and bed pressure measured by Sivakumaran (1981) and the results obtained with the three models mentioned in the previous section: the non-hydrostatic version of the VAM model obtained with the GUI, the hydrostatic version of the VAM model and the IBER model.

Table 2. Input variables for the *flow over Gaussian obstacle* module of the software platform to generate Fig. 12.

q (m^2/s)	**0.0561**	**0.07703**	**0.10144**
h_u (m)	0.35	0.35	0.35
h_d (m)	0.02	0.02	0.02
a (m)	0.2	0.2	0.2
b (m)	8.6806	8.6806	8.6806
L (m)	3.2	3.2	3.2
x_c (m)	1.6	1.6	1.6
n (s/m$^{1/3}$)	0.01	0.01	0.01
CFL	0.5	0.5	0.5
Δx (m)	0.05	0.05	0.05
T_{max} (s)	5	5	5

As depicted in Fig. 12, the non-hydrostatic results obtained with the VAM model software platform are in excellent agreement with the experimental free surface and bed pressure distributions, since the VAM model accounts for vertical flow effects. The improvements with respect to the simulations carried out with the hydrostatic VAM and IBER models are more evident on inspecting the bed pressure profiles results, which are markedly non-hydrostatic. The hydrostatic-based models consider that the bed pressure equals the water depth and, thus, are unable to reproduce the bed pressure profiles over the obstacle. Further, the higher free surface levels upstream of the obstacle obtained with the hydrostatic models result in an unrealistic prediction of the discharge coefficient of the structure (Castro-Orgaz and Hager, 2017).

Curved-shaped obstacles are instream structures used worldwide for water discharge measurement in open channels and rivers. The measurement of the discharge is conducted reading the upstream depth and applying a weir flow equation with an accurate estimation of the discharge coefficient of the structure (Castro-Orgaz and Hager, 2017). It is clearly depicted in Fig. 12 that the VAM model produces such an accurate estimation of the discharge capacity of the obstacle. These results confirm the usefulness of the developed software platform to reproduce non-hydrostatic conditions in open channel applications.

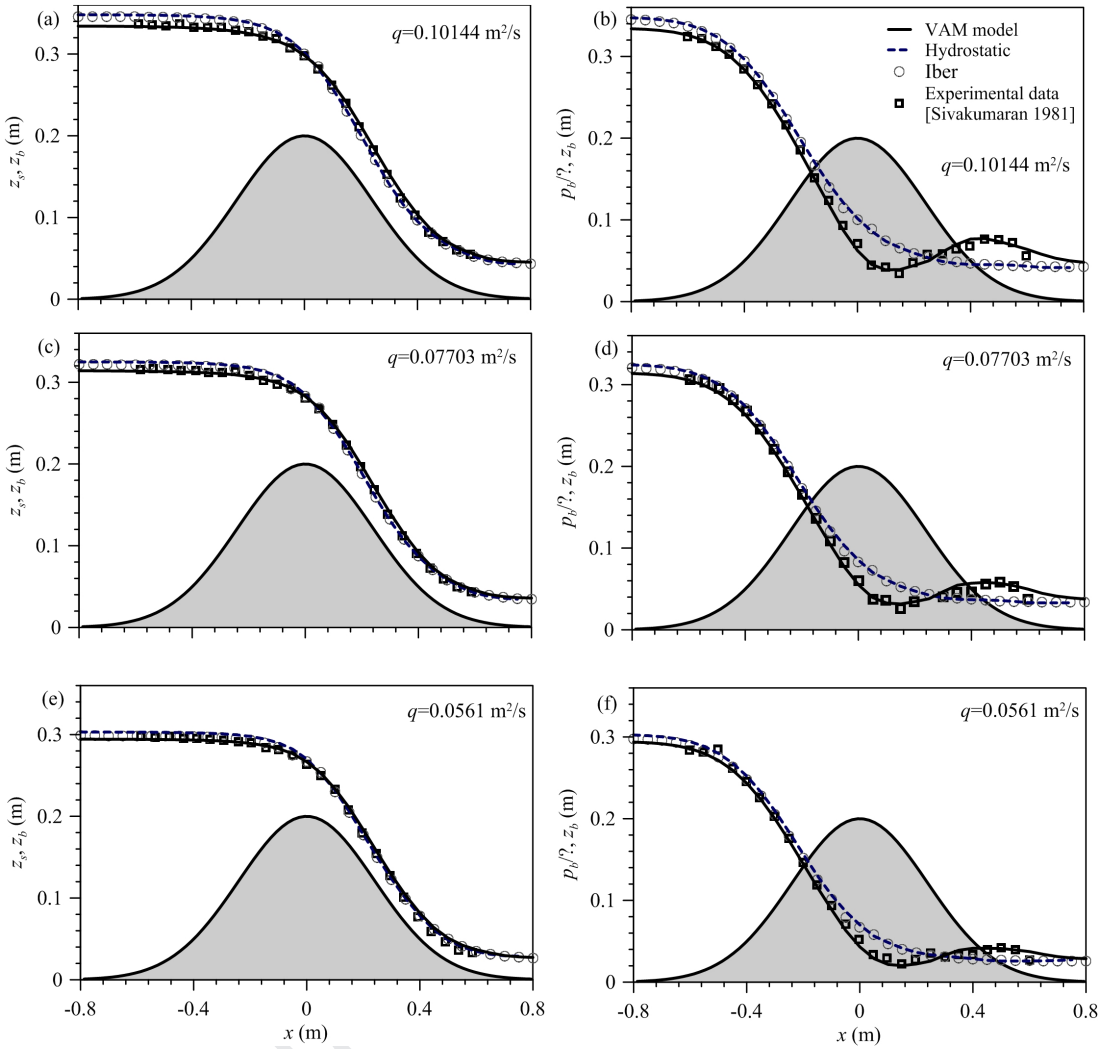


Figure 12. Free surface and bed pressure distributions for the flow over a Gaussian obstacle obtained with the hydrostatic (grey dashed line) and non-hydrostatic (black solid line) versions of the VAM model and with the IBER model (grey dots) at different times. Comparisons with experimental measurements by Sivakumaran (1981) (black squares).

5. Conclusions

In this study, a user-friendly software platform is presented to model depth-integrated non-hydrostatic flows through the **V**ertically-**A**veraged and **M**oment

(VAM) equations model. The platform comprises four applications: solitary wave propagation, solitary wave propagation with run-up, Favre waves and flow over Gaussian obstacles. After the description of the governing equations, numerical scheme and structure of the software platform, it is applied to solve several coastal and open channel flow problems.

First, the platform is used to reproduce solitary wave propagation over horizontal fixed bed (with A/h = 0.456 and h = 2.5 m), as well as to model the propagation and run-up of a solitary wave with A/h = 0.3 and h = 1 m over a plane beach. The results are compared with the hydrostatic version of the VAM model, thereby resulting that the non-hydrostatic solutions obtained with the software platform are significantly more realistic in both cases. In the case of solitary wave propagation and run-up, the software platform results are also compared to laboratory data collected by Synolakis (1986). The software outputs reproduce the experimental cross-shore distributions of the free surface at different times; including the solitary wave deformation before breaking, wave breaking and run-up.

Then, the software is used to model Favre waves for three different Froude numbers (F = 1.081, F = 1.104 and F = 1.192) and the results are compared to the hydrostatic version of the VAM model, the user-friendly hydrostatic model IBER and experimental data collected by Soares-Frazão and Zech (2002). The agreements between free-surface time series obtained with the developed software and measured free surface values are excellent; whereas the results obtained with the hydrostatic version of the VAM model and the IBER model are significantly worse.

Finally, the flow over a Gaussian obstacle is simulated with the three aforementioned models for three unit discharge runs: q = 0.0561 m^2/s, q = 0.07703 m^2/s and q = 0.10144 m^2/s. The computational results are compared with free surface and bed pressure distributions measured in a laboratory facility by Sivakumaran (1981). Again, the fits to the experimental measurements provided by the developed

VAM model software platform are better than those obtained with hydrostatic modelling approaches.

The implementation of a numerical solver for the VAM equations is time-consuming and not evident, and made most of the scientific community working in vertically-integrated non-hydrostatic flows to tend away from solving them. However, when the VAM model is properly structured and discretized, a useful MATLAB application was produced. It permits the VAM platform user to solve this complex system of equations (as compared to Saint Venant equations) in an easy and accurate way, thereby producing computational results which may be used to test other models, like those based on Boussinesq equations.

The results of this work remark the importance of the VAM model not only for modelling non-hydrostatic flows in open channel tests, but also for reproducing nearshore flows and coastal processes. This model can be used for a wide range of applications, such as the prediction of free surface profiles at the vicinity of bed forms and obstacles; the water discharge measurement in open channels and rivers, the estimation of wave run-up in coastal areas, which is the key variable to delimit the public domain; or the design of laboratory experiments in wave and hydraulic flumes, among others. The GUI presented in this paper can be helpful for students, professors and water managers, whereas the MATLAB routines of the VAM model provided as supplementary material are intended for engineers and scientists doing research in this area.

Acknowledgements

This work was supported by the research project CTM2017-85171-C2-1-R (Secretaría de Estado de I+D+i, Spain) and the research group AGR-127 (Junta de Andalucía). RB and FNCC were partly funded by the Spanish Ministry of Science, Innovation and Universities through Programa Juan de la Cierva 2017 (FJCI-2017-

31781) and Programa Juan de la Cierva 2016 (FJCI-2016-28009), respectively. We thank four anonymous reviewers for their improvements to this work.

References

Aissiouene, N., Bristeau, M. O., Godlewski, E., & Sainte-Marie, J. (2015). A combined finite volume-finite element scheme for a dispersive shallow water system. Netw. Heterogen Media, 11, 1-27.

Aureli, F., Maranzoni, A., Mignosa, P., & Ziveri, C. (2008). A weighted surface-depth gradient method for the numerical integration of the 2D shallow water equations with topography. Advances in Water Resources, 31(7), 962−974.

Bai, Y., & Cheung, K. F. (2013). Dispersion and nonlinearity of multi-layer non hydrostatic free-surface flow. Journal of Fluid Mechanics 726, 226−260.

Bladé, E., Cea, L., Corestein, G., Escolano, E., Puertas, J., Vázquez-Cendón, E., Dolz, J., & Coll, A. (2014). Iber: herramienta de simulación numérica del flujo en ríos. Revista Internacional de Métodos Numéricos para Cálculo y Diseño en Ingeniería, 30(1), 1−10.

Brocchini, M., & Peregrine, D. H. (1996). Integral flow properties of the swash zone and averaging. Journal of Fluid Mechanics 317, 241−273.

Brocchini, M., 2013. A reasoned overview on Boussinesq-type models: the interplay between physics, mathematics and numerics. Proceedings of the Royal Society A: Mathematical, Physical and Engineering Sciences 469, 20130496.

Cea, L., & Bladé, E. (2015). A simple and efficient unstructured finite volume scheme for solving the shallow water equations in overland flow applications. Water Resources Research, 51(7), 5464−5486.

Cea, L., Bermudez, M., Puertas, J., Blade, E., Corestein, G., Escolano, E., Conde, A., Bockelmann-Evans, B., & Ahmadian, R. (2016). IberWQ: New simulation tool

583 for 2D water quality modelling in rivers and shallow estuaries. Journal of
584 Hydroinformatics, 18(5), 816−830.

585 Cantero-Chinchilla, F. N., Castro-Orgaz, O., Dey, S., & Ayuso, J. L. (2016).
586 Nonhydrostatic dam break flows. I: Physical equations and numerical schemes.
587 Journal of Hydraulic Engineering, 142(12), 04016068.

588 Cantero-Chinchilla, F. N., Castro-Orgaz, O., & Khan, A. A. (2018a). Depth-
589 integrated nonhydrostatic free-surface flow modeling using weighted-averaged
590 equations. International Journal for Numerical Methods in Fluids, 87(1), 27−50.

591 Cantero-Chinchilla, F. N., Castro-Orgaz, O., Schmocker, L., Hager, W. H., & Dey, S.
592 (2018b). Depth-averaged modelling of granular dike overtopping. Journal of
593 Hydraulic Research, 56(4), 537−550.

594 Casulli, V. A. (1999). A semi-implicit finite difference method for non-hydrostatic,
595 free-surface flows. International Journal for Numerical Methods in Fluids 30(4),
596 425−440.

597 Castro-Orgaz, O., & Hager, W.H. (2017). Non-hydrostatic free surface flows.
598 Advances in Geophysical and Environmental Mechanics and Mathematics.
599 Springer, Berlin.

600 Escalante, C., de Luna, T. M., & Castro, M. J. (2018). Non-hydrostatic pressure
601 shallow flows: GPU implementation using finite volume and finite difference
602 scheme. Applied Mathematics and Computation, 338, 631-659.

603 Escalante, C., Fernández-Nieto, E. D., de Luna, T. M., & Castro, M. J. (2019). An
604 efficient two-layer non-hydrostatic approach for dispersive water waves. Journal
605 of Scientific Computing, 79(1), 273−320.

606 Favre H. (1935). Etude theorique et experimentale des ondes de translation dans les
607 canaux decouverts, Dunod, Paris.

608 Fernandez-Nieto, E. D., Parisot, M., Penel, Y., & Sainte-Marie, J. (2018). A hierarchy
609 of dispersive layer-averaged approximations of Euler equations for free surface
610 flows. Communications in Mathematical Sciences, 16(5), 1169-1202.

611 Khan, A. A., & Steffler, P. M. (1996a). Vertically averaged and moment equations
612 model for flow over curved beds. Journal of Hydraulic Engineering 122(1), 3−9.

613 Khan, A. A., & Steffler, P. M. (1996b). Modeling overfalls using vertically averaged
614 and moment equations. Journal of Hydraulic Engineering 122(7), 397−402.

615 Kim, D. H., Lynett, P. J., & Socolofsky, S. A. (2009). A depth-integrated model for
616 weakly dispersive, turbulent, and rotational fluid flows. Ocean
617 Modelling 27(3−4), 198−214.

618 Kim, D. H., & Lynett, P. J. (2011). Dispersive and nonhydrostatic pressure effects at
619 the front of surge. Journal of Hydraulic Engineering 137(7), 754−765.

620 Madsen, P. A., Murray, R., & Sørensen, O. R. (1991). A new form of the Boussinesq
621 equations with improved linear dispersion characteristics. Coastal
622 engineering, 15(4), 371-388.

623 Madsen, P. A., & Sørensen, O. R. (1992). A new form of the Boussinesq equations
624 with improved linear dispersion characteristics. Part 2. A slowly-varying
625 bathymetry. Coastal engineering, 18(3-4), 183-204.

626 Nadiga, B. T., Margolin, L. G., & Smolarkiewicz, P. K. (1996). Different
627 approximations of shallow fluid flow over an obstacle. Physics of Fluids, 8(8),
628 2066−2077.

629 Peregrine, D. H. (1966). Calculations of the development of an undular bore. Journal
630 of Fluid Mechanics, 25(2), 321−330.

631 Sivakumaran, N.S. (1981). Shallow-flow over curved beds. Doctoral Thesis, Asian
632 Institute of Technology, Bangkok, Thailand.

633 Soares-Frazao, S., & Zech, Y. (2002). Undular bores and secondary waves-
634 Experiments and hybrid finite-volume modelling. Journal of hydraulic
635 research, 40(1), 33−43.

636 Soares-Frazão, S., & Guinot, V. (2008). A second-order semi-implicit hybrid scheme
637 for one-dimensional Boussinesq-type waves in rectangular channels. International
638 journal for numerical methods in fluids, 58(3), 237−261.

639 Stansby, P. K., & Zhou, J. G. (1998). Shallow-water flow solver with non-hydrostatic
640 pressure: 2D vertical plane problems. International Journal of Numerical Methods
641 in Fluids 28, 541−563.

642 Steffler, P. M., & Jin, Y. C. (1993). Depth averaged and moment equations for
643 moderately shallow free surface flow. Journal of Hydraulic Research 31(1), 5−17.

644 Tavakkol, S., & Lynett, P. (2017). Celeris: A GPU-accelerated open source software
645 with a Boussinesq-type wave solver for real-time interactive simulation and
646 visualization. Computer Physics Communications, 217, 117-127.

647 Teng, J., Jakeman, A. J., Vaze, J., Croke, B. F., Dutta, D., & Kim, S. (2017). Flood
648 inundation modelling: A review of methods, recent advances and uncertainty
649 analysis. Environmental Modelling& Software, 90, 201−216.

650 Toro, E. F. (2001). Shock-capturing methods for free-surface shallow flows.
651 Singapore: John Wiley & Sons.

652 Toro, E. F. (2009). Riemann solvers and numerical methods for fluid dynamics: a
653 practical introduction. Berlin: Springer.

654 U.S. Corps of Engineers (2002). HEC-RAS River Analysis System, Hydraulic
655 Reference Manual. Hydraulic Engineering Center Report CPD-69, Davis, CA.

656 Warren, I. R., & Bach, H. (1992). MIKE 21: a modelling system for estuaries, coastal
657 waters and seas. Environmental Software, 7(4), 229−240.

Wei, Z., & Jia, Y. (2013). A depth-integrated non-hydrostatic finite element model for wave propagation. International Journal for Numerical Methods in Fluids 73(11), 976−1000.

Wei, Z., & Jia, Y. (2014a). Non-hydrostatic finite element model for coastal wave processes. Coastal Engineering 92, 31−47.

Wei, Z., & Jia, Y. (2014b). Simulation of nearshore wave processes by a depth-integrated non-hydrostatic finite element model. Coastal Engineering 83, 93−107.

Yamamoto, S., & Daiguji, H. (1993). Higher-order-accurate upwind schemes for solving the compressible Euler and Navier-Stokes equations. Computers & Fluids 22(2−3), 259−270.

Yamazaki, Y., Kowalik, Z., & Cheung, K. F. (2009). Depth-integrated, non-hydrostatic model for wave breaking and run-up. International Journal for Numerical Methods in Fluids 61(5), 473−497.

Zijlema, M., & Stelling, G. S. (2005). Further experiences with computing non hydrostatic free-surface flows involving water waves. International Journal for Numerical Methods in Fluids 48(2), 169−197.

Zijlema, M., & Stelling, G. S. (2008). Efficient computation of surf zone waves using the nonlinear shallow water equations with non-hydrostatic pressure. Coastal Engineering 55(10), 780−790.

HIGHLIGHTS

- A software platform to compute depth-integrated non-hydrostatic flows is presented.
- The software is based on the Vertically-Averaged and Moment equations model.
- The software is applied to solve several open channel and coastal flow tests.
- Excellent agreements are obtained between the software results and experimental data.

Declaration of interests

☒ The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

☐The authors declare the following financial interests/personal relationships which may be considered as potential competing interests:
