Communications on Applied Nonlinear Analysis
ISSN: 1074-133X
Vol 32 No. 9s (2025)

Artificial Viscosity Stabilization of Meshless Methods for Shallow Water Flows in Channels

Mariam Ijerch[1], Mohamed Sadik[2]

[1]Engineering Sciences Laboratory, Faculty of Sciences, Agadir, Ibn Zohr University, Morocco

(Email: mariam.ijerch@edu.uiz.ac.ma)

[2]Engineering Sciences Laboratory, Faculty of Sciences, Agadir, Ibn Zohr University, Morocco

(Email: m.sadik@uiz.ac.ma)

Article History:

Received: 12-01-2025

Revised: 15-02-2025

Accepted: 01-03-2025

Abstract:
The purpose of this paper is to present an efficient localized meshless methods based on radial basis functions to accurately analyze the shallow water equations (SWEs) in open channels. It is a hyperbolic system of first-order nonlinear partial differential equations. Due to their classification as advection equations, they allow for discontinuous solutions, often characterized by shock behaviour in simulations. Therefore, the development of an efficient and accurate numerical model for analyzing the SWEs holds critical importance in scientific research. To address non-physical oscillations near discontinuities, we developed a technique involving artificial viscosity, which is integrated with the local meshless methods for spatial discretization of the SWEs, while temporal discretization is achieved using the fourth-order Runge-Kutta method. A series of experiments has been conducted to evaluate the effectiveness and accuracy of the suggested methods. These experiments included examining flow at a steady state with and without friction, as well as studying the dam-break scenario on a wet bed. The results compared with analytical solutions, demonstrate that the artificial viscosity combined with the proposed methods proficiently capture shocks and effectively handles discontinuous flow by introducing an appropriate viscosity coefficient into the equations. Overall, the results are satisfactory and show good agreement.

Keywords: Shallow water equations, Open channel, Radial basis function partition of unity method (RBF-PUM), Radial basis function finite difference (RBF-FD) method, QR factorization.

1. Introduction

The shallow water equations (SWEs) were first proposed in 1871 by Saint-Venant [1] to describe various applications in ocean and hydraulic engineering, such as coastal flow, open-channel flow in natural rivers and prismatic channels, flood waves, failure of dams, etc. Under some essential assumptions [2], the SWEs are derived by three-dimensional Navier-Stokes equations where the horizontal wave length is much larger than the depth of the fluid. SWEs form a system of non-linear hyperbolic partial differential equations (PDEs) governed by conservation of mass and balance of momentum. The critical problem in solving this system of equations is the spurious oscillation happens in the discontinuous areas such as shock waves, dam break, or hydraulic jump problems. The analytical solution for these problems is only available for a limited number of special cases [3]. Therefore it is significant to develop a reliable and accurate numerical solver.

In the past decades, plenty of mesh-based numerical models have been developed for solving the SWEs, such as the Finite Difference Method (FDM) [4], Finite Element Method (FEM) [5], and Finite Volume Method (FVM) [6, 7]. However, for multidimensional problems defined on a complex geometries, mesh generation have a common difficulty that it is a time-consuming and their performance depends on the quality of the mesh. To overcome these difficulties, many researchers have paid attention to use the newly developed meshless method as the promising alternative to traditional methods which have been employed to analyse various problems. The meshless methods were also adopted to solve SWEs by many scholars. Among them, the development of meshless methods based on Radial Basis Functions (RBFs), such as multiquadric method [8], Compactly Supported RBF (CSRBF) [9], Local Radial Basis Function based Differential Quadrature Method (LRBFDQM) [10], Extrapolated Local Radial Basis Functions Collocation Method (ELRBFCM) [11], and others. The main advantages of RBF-based methods for solving partial differential equations lie in their simplicity, flexibility, and applicability to various PDEs. They are particularly effective in handling high-dimensional problems with complex geometries.

Since the shallow water equations are a hyperbolic non-linear partial differential equations, the information of characteristic and the correct directions of wave transmission are quite important to numerical simulation. The first author who introduced the radial basis functions method in the field of PDEs was Kansa in 1990 [12, 13]. He developed the global RBF method which is a numerical method that gives an approximate solution for PDEs. It can be considered as one of the collocation methods which are efficient and expected to produce spectral accuracy for the numerical solution. However, the infinitely smooth RBFs (Table 1) produce dense matrices thus ill-conditioned matrices, especially for the parameter values that lead to the highest accuracy. To overcome this issue, it is essential to introduce certain locality in the system to be solved. Not only the conditioning will be improved, but also the accuracy of the numerical solution will be guaranteed. In this context, we conducted our studies using local and stable meshless methods in addition to Kansa's method to solve SWEs. One such method is the RBF Finite Difference method (RBF-FD), first described in [14] in 2000. The idea of this method is to form computational stencils, similar to those in finite difference method, by constructing derivative discretizations based on RBF interpolants on localized node sets [15]. Another localized method is the RBF Partition of Unity Method (RBF-PUM) which is based on approximating the solution of PDEs in a regular covering of overlapping subdomains and then patching up the local solutions using the partition of unity principle [16, 17]. The RBF-PUM, increasing the sparsity of linear systems, enables us to offer significant savings in the computational time, but in order to guarantee the stability of the solution, as $\varepsilon \longrightarrow 0$ (called flat limit regime), a stable evaluation algorithm is employed. The stable RBF-QR algorithm bypasses some limitations, including complications related to the selection of ε, ill-conditioning of RBF-PUM matrices. The main logic behind the RBF-QR method is to replace the bad basis (ill-conditioned) with a good one that can span the same space [18, 19].

The RBF methods that we investigate are: the RBF generated finite difference(RBF-FD) method, the RBF partition of unity method (RBF-PUM) and Kansa's method. The first two methods are localized such that the final system of equations is sparse, and the third method is global and gives a dense system of equations. It is known, based on numerical experiments, that all three methods in general are unstable when computing collocated solutions to time-dependent conservation laws without

additional stabilization [20]. In [21], the author suggests activating artificial viscosity in the time domain rather than in localized regions of space and using hyperviscosity provides an elegant approach to spatially selecting the diffusion effect based on supplementing the numerical scheme by a hyperviscosity term. Hyperviscosity was first introduced to spectral methods in 1998 by Ma under the name (super)spectral viscosity [22]. Despite the promising analytical results demonstrated by Ma, the method was never widely adopted and has only been mentioned in a few research papers where it was applied to the Navier-Stokes equations. The meshless community first adopted spectral viscosity, now called hyperviscosity, in 2011 when Fornberg [20] demonstrated that hyperviscosity could stabilize time-stepping schemes by shifting spurious eigenvalues into the stable region. Subsequently, hyperviscosity was applied to a variety of fluid dynamics cases [23, 24]. Initially, the parameters of hyperviscosity were only briefly mentioned and included a large number of empirical approximations.

Recently, new artificial hyperviscosity formulations have been proposed in the literature [25], which are very effective for RBF-FD methods for solving surface convection-diffusion equations and for nonlinear conservation laws [26]. additionally, meshless schemes combined with hyperviscosity technique have been proposed to the authors' knowledge such in [27] used the RBF-based method for solving the shallow water equations to capture shocks. Moreover, hyperviscosity and its connection to the stability of the RBF Partition of Unity (RBF-PU) method was only well established by the work of Liu et al. [28], who showed that the hyperviscosity improve the stability of the RBF-PU method for solving convection-diffusion equations on surfaces. Therefore, we suggested to combine the RBF-FD (using Inverse Multiquadratic function) and RBF-PUM-QR with hyperviscosty technique in order to establish a localized numerical meshless model which can accurately and effectively solve the SWEs.

This paper is organized as follows. In Section 2, the governing equations and some properties are explained. The Kansa's method, RBF-FD method and RBF-PUM with QR factorization with Hyperviscosity technique is presented in Section 3. Then, two challenging tests are adopted to evaluate the accuracy and stability of the numerical methods in Section 4. Finally, some conclusions are given in Section 5.

2. Governing equations and properties

Saint-Venant [1] introduced the mathematical model for shallow water, known as Saint-Venant's Equations or Shallow Water Equations (SWEs). These equations are based on fundamental physical principles, specifically the laws of mass conservation and momentum conservation. The model is applicable to free surface flows when the horizontal scales of water mass significantly exceed the vertical scale and the flow in the vertical direction is negligible. The water density in our model is considered to be constant and Coriolis force is neglected. As the source terms the bed slope term and bed friction term are considered. The equations of interest consist of the following system:

$$\begin{cases} \partial_t h + \partial_x q = 0, \\ \partial_t q + \partial_x \left(\frac{q^2}{h} + \frac{1}{2} g h^2 \right) = g h S_b - S_f, \end{cases} \tag{1}$$

where $h(x,t)$ is the height of the water, $u(x,t)$ is the flow velocity, $q(x,t) = h(x,t)u(x,t)$ is the flow discharge and g is the acceleration due to gravity. Concerning the source terms, $S_b = -\partial_x b$ is the bed slope where $b: \mathbb{R} \to \mathbb{R}^+$ is a given smooth function which describes the topography and S_f is

the friction term, it is a given function of h and u, two examples widely used in hydrology are the Manning and the Darcy-Weisbach friction laws, which are given respectively by

$$S_f = C_f g \frac{u|u|}{h^{\frac{1}{3}}} = C_f g \frac{q|q|}{h^{\frac{7}{3}}}, \tag{2}$$

where $C_f = n^2$, n is Manning's coefficient.

$$S_f = C_f g u|u| = C_f g \frac{q|q|}{h^2}, \tag{3}$$

where $C_f = \frac{f}{8g}$, f is dimensionless coefficient. In order to emphasize the properties of (1) without friction, we rewrite the one-dimensional equations in quasilinear form

$$\partial_t \mathbf{Q} + \mathrm{A}(\mathbf{Q})\, \partial_x \mathrm{Q} = \mathrm{S}(\mathbf{Q}), \tag{4}$$

with

$$\boldsymbol{Q} = \begin{pmatrix} h \\ q \end{pmatrix}, \qquad \mathrm{A}(\mathbf{Q}) = \begin{pmatrix} 0 & 1 \\ -u^2 + gh & 2u \end{pmatrix}, \qquad S(\boldsymbol{Q}) = \begin{pmatrix} 0 \\ -\mathrm{gh}\, \partial_x \mathrm{b} \end{pmatrix}. \tag{5}$$

When $h > 0$, the system is strict hyperbolic with the eigenvalues of A are given by

$$\lambda_1 = u - \sqrt{gh} \qquad and \qquad \lambda_2 = u + \sqrt{gh}. \tag{6}$$

These eigenvalues represent the wave speeds which are basic characteristics of the flow. When $h = 0$ (dry zone) the eigenvalues coincide and the system looses hyperbolicity. For considering the well-posed problem in 1D described by the SWEs, the number of boundary conditions should be confirmed by the Froude number

$$\mathrm{Fr} = \frac{|u|}{\sqrt{gh}}. \tag{7}$$

If Fr < 1 the flow is considered subcritical, indicating the presence of one positive and one negative eigenvalue. In such cases, it is necessary to define one boundary value on the left and another on the right. On the other hand, when Fr > 1 then the flow is supercritical which means that two values have to be specified at one of the boundaries depending on the sign of the velocity. A transcritical regime can exist in the solution if parts of the flow are subcritical and other parts are supercritical.

2.1 Hyperviscosity technique

Although the global stability theory of RBF-FD and RBF-PUM methods is largely undeveloped, Fornberg and Lehto [20] proposed a practical method to correct the spectra of RBF-FD differentiation matrices for hyperbolic operators on the sphere by introducing artificial hyperviscosity. Their approach effectively shifted rogue eigenvalues to the left half of the complex plane. This issue is also caused by using large RBF-FD stencils, where increasing the stencil size n_{loc} enhances the accuracy of the approximation. Based on the above analysis, we will use the Kansa, RBF-FD, and RBF-PUM methods combined with an artificial hyperviscosity formulation to solve the shallow water equations, where the convection term dominates and leads to non-physical oscillations that would otherwise amplify with time.

Stabilization of the methods is achieved by applying a hyperviscosity as a filter to the right hand side of the model, the equation solved takes the form

$$\partial_t \boldsymbol{Q} = -F(\boldsymbol{Q}) + S(\boldsymbol{Q}) + H(\boldsymbol{Q}), \tag{8}$$

where

$$\boldsymbol{Q} = \begin{pmatrix} h \\ q \end{pmatrix}, \qquad \mathrm{F}(\mathbf{Q}) = \begin{pmatrix} \partial_x q \\ \partial_x\left(\frac{q^2}{h} + \frac{1}{2}gh^2\right) \end{pmatrix}, \qquad S(\boldsymbol{Q}) = \begin{pmatrix} 0 \\ ghS_b - S_f \end{pmatrix}, \tag{9}$$

and H is the hyperviscosity filter defined by

$$H = \mu \frac{\partial^{2k}}{\partial x^{2k}}, \tag{10}$$

where $\mu \in R$ and $k \in N^*$ small numbers that must be selected. The constant μ has to be selected carefully to provide enough stabilisation without ruining the solution [20, 23]. There is no broad consensus in the literature with an abundance of proposed estimates and scalings for the constant μ. However, there are many studies suggesting that the hyperviscosity parameter should be chosen through trial and error. Fornberg and Lehto [20] provide suggestions for selecting μ in the RBF-FD method, they found scaling $\mu \sim N^{-2k}$ worked well, where N is the total number of nodes in the domain. Then, in [23], Flyer et al. empirically computed μ and k for the shallow water equations on the sphere. They found experimentally that for given values of N, n_{loc} and ε, choosing $\mu = \mu_c N^{-k}$ where μ_c ranging from O(1) to O(10^{-2}) provides stability with good accuracy for PDEs.

In this paper, we experimentally determine the hyperviscosity parameter through trial and error, the order k set to one, drawing on the work of Dehghan and Abbaszadeh [29], who used a value of 10^{-3} to stabilize their method for the dam break problem. We also reference [27], where values ranging from 0.01 to 0.5 were employed with an order k set to one, and [30], where a small value around 10^{-5} was deliberately chosen with the same order k.

3. Numerical methods

3.1 Kansa method

First, we introduce the well-known Kansa method [12, 13]. It is also referred to as Radial Basis Function Collocation Method (RBFCM). The main idea of the method is approximating the solution in terms of linear combination of infinitely differentiable Radial Basis Functions (RBFs) φ which are listed in Table 1, this functions depends only on the distance to a center point and shape parameter ε. This method is particularly useful for solving PDEs in complex geometries with irregularly distributed points, offering an efficient and flexible numerical technique. However, The main disadvantage of the method is that it represents the discretization of the PDE as a dense matrix. These matrices are

extremely sensitive to the choice of the shape parameter for the RBFs, which makes it difficult to solve problems with a large number of unknowns. This occurs because in the use of RBF interpolation and add more nodes, the matrices involved become less stable. This is especially true when a poor choice has been made when designating the RBF centers, and also when infinitely smooth basis functions are used with associated shape parameters set to extreme values. To have more details on the theoretical background, we refer the reader to [35, 36].

Table 1. Infinitely smooth RBFs

Multiquadric	Inverse multiquadratic	Inverse quadratic	Gaussian
$(\mathbf{1}+\boldsymbol{\epsilon}^{\mathbf{2}}\boldsymbol{r}^{\mathbf{2}})^{\frac{\mathbf{1}}{\mathbf{2}}}$	$(\mathbf{1}+\boldsymbol{\epsilon}^{\mathbf{2}}\boldsymbol{r}^{\mathbf{2}})^{\frac{-\mathbf{1}}{\mathbf{2}}}$	$(\mathbf{1}+\boldsymbol{\epsilon}^{\mathbf{2}}\boldsymbol{r}^{\mathbf{2}})^{-\mathbf{1}}$	$\boldsymbol{e}^{-\epsilon^2 r^2}$

3.1.1 Method's principle

In this section, we introduce the general notation and quantities we need for RBF approximation of SWEs. Assume that $X = \{x_1, x_2, \ldots, x_N\}$ is a set of distinct points in $\Omega \subset \mathbb{R}^d$. We start from given an interpolant I_u that approximates a function u:

$$I_u(x) = \sum_{i=1}^{n} \lambda_i \, \varphi(\|x - x_i\|), \qquad x \in \Omega, \tag{11}$$

where $\lambda_i \in \mathbb{R}$ are the unknown coefficient to be determined and φ can be any radial basis function such in Table 1, the shape parameter ε appearing in the RBFs dictates the flatness of the radial basis function and plays a key role in the convergence rate of the approximations and the condition number of the coefficient matrices.
By applying the interpolation criteria $I_u(x_j) = u(x_j)$, the coefficient λ_j , $j = 1, \ldots, N$ is discovered. And we obtain a linear system as follow:

$$A\bar{\lambda} = U, \tag{12}$$

where

$$A = \begin{pmatrix} \varphi(\|x_1 - x_1\|) & \cdots & \varphi(\|x_1 - x_N\|) \\ \vdots & \ddots & \vdots \\ \varphi(\|x_N - x_1\|) & \cdots & \varphi(\|x_N - x_N\|) \end{pmatrix}, \tag{13}$$

$$\bar{\lambda} = [\lambda_1, \lambda_2, \ldots, \lambda_N]^T, \qquad U = [u(x_1), u(x_2), \ldots, u(x_N)]^T. \tag{14}$$

When solving PDE, we prefer to work with the discrete approximate instead of the coefficients. From (11) and (12) we can write

$$I_u(x) = \bar{\varphi}(x) A^{-1} U = D(x) U, \tag{15}$$

where $\bar{\varphi}(x) = [\varphi(\| x - x_1 \|), \varphi(\| x - x_2 \|), \dots, \varphi(\| x - x_N \|)]$ and $D = \bar{\varphi}A^{-1}$. Micchlli [37] proved that the strict positive definiteness of an RBF listed in Table 1 guarantees that the interpolation matrix A in (13) is also positive definite, and thus invertible. Because our final target is to solve PDEs, we apply a linear differential operator L to the RBF approximation, we get

$$(I_u)_L(x) = \bar{\varphi}_L(x)A^{-1}U = D_L(x)U, \qquad (16)$$

where $D_L = \bar{\varphi}_L A^{-1}$ is known as a differentiation matrix.

3.1.2 Spatial discretization of SWEs using Kansa method

In this section, we will describe the implementation of Kansa method for shallow water equations. Let $\{x_1, x_2, \dots, x_N\}$ be a set of scattered nodes considered in the study area. Using (16), the first and second spatial derivatives of a function $\boldsymbol{Q}(x,t)$ can be expressed as follows:

$$\frac{\partial \boldsymbol{Q}(x,t)}{\partial x} = D_x \boldsymbol{Q}(x,t), \qquad \frac{\partial^2 \boldsymbol{Q}(x,t)}{\partial x^2} = D_{xx}\boldsymbol{Q}(x,t), \qquad (17)$$

where D_x and D_{xx} are first and second derivative matrices with respect to x. The system (8) is approximated by the following equivalent system:

$$\partial_t \boldsymbol{Q} = -\bar{F}(\boldsymbol{Q}) + \bar{S}(\boldsymbol{Q}) + \bar{H}(\boldsymbol{Q}), \qquad (18)$$

where $\bar{F}$ is the convective flux, $\bar{H}$ contains the hyperviscosity and $\bar{S}$ is the source term due to bottom elevator. Using the Kansa method, we obtain

$$\bar{F}(\boldsymbol{Q}) = \begin{pmatrix} D_x q \\ D_x\left(\frac{q^2}{h} + \frac{1}{2}gh^2\right) \end{pmatrix}, \quad \bar{S}(\boldsymbol{Q}) = \begin{pmatrix} 0 \\ -ghD_x b \end{pmatrix}, \quad \bar{H}(\boldsymbol{Q}) = \begin{pmatrix} \mu D_{xx} h \\ \mu D_{xx} q \end{pmatrix}, \qquad (19)$$

Equivalent to

$$\partial_t \boldsymbol{Q} = RHS_{kansa}(\boldsymbol{Q}) \quad \text{where} \quad RHS_{kansa}(\boldsymbol{Q}) = -\bar{F}(\boldsymbol{Q}) + \bar{S}(\boldsymbol{Q}) + \bar{H}(\boldsymbol{Q}). \qquad (20)$$

3.2 Radial Basis Function Finite Difference Method (RBF-FD)

First introduced by Tolstykh [14], RBF-FD formulas are derived through RBF interpolation over local sets of nodes on the surface. This type of method is conceptually similar to the standard Finite Difference (FD). The fundamental principle underlying RBF-FD revolves around the approximation of the differential operator for solutions at interior nodes. The approximation is achieved by means of a linear combination of function values at the neighbouring node locations and then determining the RBF-FD weights by assuming that the approximations become exact for all RBFs that are centred at the neighbouring nodes, as opposed to the standard FD, which focuses on approximating polynomials for the same nodes sets. After performing calculations at all interior nodes, the approximate solution

can be computed from the associated linear system of equations, for which the RBF-FD system matrix is sparse, thus can be effectively inverted.

3.2.1 Method's principle

We will next present an outline of the RBF along with finite-difference (RBF-FD) formulation. Let $X = \{x_1, x_2, \ldots, x_N\}$ is a set of distinct points in $\Omega \subset \mathbb{R}^d$ and let $I(x_j) = \{x_1^j, x_2^j, \ldots, x_{n_{loc}}^j\}$ be a set of points to form a stencil weight at x_j . Note that the number of points n_{loc} in each stencil is either constant or vary with j, without loss of generality we suppose that n_{loc} is constant. In RBF-FD, any linear differential operator L that acts on $u(x)$ evaluated at x_j can be approximated by a linear weighted combination of the function values at the points of $I(x_j)$, i.e.

$$Lu(x_j) \approx \sum_{k=1}^{n_{loc}} \omega_k^{(j)} u(x_k^{(j)}). \qquad (21)$$

The RBF-FD weights $\omega_k^{(j)}$, $k = 1, \ldots, n_{loc}$ are computed by assuming the approximations (21) become exact for all RBFs φ (Table 1) that are centred at the nodes $x_i^{(j)}$, $i = 1, \ldots, n_{loc}$, i.e., we assume that (21) becomes exact for function $u(x_j) = \varphi(\| x - x_i^{(j)} \|)\Big|_{x=x_j}$, $i = 1, \ldots, n_{loc}$, this assumption leads to $n_{loc} \times n_{loc}$ linear system:

$$\begin{pmatrix} \varphi\left(\left\|x_1^{(j)} - x_1^{(j)}\right\|\right) & \cdots & \varphi\left(\left\|x_1^{(j)} - x_{n_{loc}}^{(j)}\right\|\right) \\ \vdots & \ddots & \vdots \\ \varphi\left(\left\|x_{n_{loc}}^{(j)} - x_1^{(j)}\right\|\right) & \cdots & \varphi\left(\left\|x_{n_{loc}}^{(j)} - x_{n_{loc}}^{(j)}\right\|\right) \end{pmatrix} \begin{pmatrix} \omega_1^{(j)} \\ \vdots \\ \omega_{n_{loc}}^{(j)} \end{pmatrix} = \begin{pmatrix} L\varphi\left(\|x - x_1^{(j)}\|\right)\Big|_{x=x_j} \\ \vdots \\ L\varphi\left(\|x - x_{n_{loc}}^{(j)}\|\right)\Big|_{x=x_j} \end{pmatrix} (22)$$

Note that the system matrix is invertible due to the properties of the RBFs mentioned before. Therefore, the RBF-FD weights can always be calculated. To obtain the RBF-FD weights, we have to solve the system (22) for each stencil center x_j , $j = 1, \ldots, N$ to form N rows of the differentiation matrix denoted by W_L where L refer to a linear differential operator that acts on solutions, W_L is a sparse matrix and contains n_{loc} non-zero elements per row. When solving PDE, we apply a similar procedure of what has done for Kansa method. After obtaining the weights, we use the approximation equation (21) to conclude an approximate solution for the PDE.

3.2.2 Spatial discretization of SWEs using RBF-FD method

In this section, we will present implementation details on the computational of RBF-FD formulation for solving equations (1). Let $\{x_1, x_2, \ldots, x_N\}$ be a set of scattered nodes. Using (21), the first and second spatial derivatives of a function $\boldsymbol{Q}(x, t)$ can be expressed as follows:

$$\frac{\partial \boldsymbol{Q}(x_j,t)}{\partial x} = \sum_{k=1}^{n_{loc}} \frac{\partial\, \omega_k^{(j)}}{\partial x} \boldsymbol{Q}\left(x_k^{(j)}, t\right), \quad \frac{\partial^2 \boldsymbol{Q}(x_j,t)}{\partial x^2} = \sum_{k=1}^{n_{loc}} \frac{\partial^2 \omega_k^{(j)}}{\partial x^2} \boldsymbol{Q}\left(x_k^{(j)}, t\right), \; j = 1, \ldots, N \quad (23)$$

After collocating the points, we obtain

$$\frac{\partial \boldsymbol{Q}(x,t)}{\partial x} = W_x \boldsymbol{Q}(x,t), \qquad \frac{\partial^2 \boldsymbol{Q}(x,t)}{\partial x^2} = W_{xx} \boldsymbol{Q}(x,t), \tag{24}$$

where W_x and W_{xx} are first and second derivative matrices with respect to x. The system (8) is approximated by the following equivalent system:

$$\partial_t \boldsymbol{Q} = -\bar{F}(\boldsymbol{Q}) + \bar{S}(\boldsymbol{Q}) + \bar{H}(\boldsymbol{Q}), \tag{25}$$

Using the RBF-FD method to perform numerical evaluation, we obtain

$$\bar{F}(\boldsymbol{Q}) = \begin{pmatrix} W_x q \\ W_x(\frac{q^2}{h} + \frac{1}{2} g h^2) \end{pmatrix}, \qquad \bar{S}(\boldsymbol{Q}) = \begin{pmatrix} 0 \\ -ghW_x b \end{pmatrix}, \qquad \bar{H}(\boldsymbol{Q}) = \begin{pmatrix} \mu W_{xx} h \\ \mu W_{xx} q \end{pmatrix}, \tag{26}$$

Equivalent to

$$\partial_t \boldsymbol{Q} = RHS_{RBF-FD}(\boldsymbol{Q}), \text{ where } RHS_{RBF-FD}(\boldsymbol{Q}) = -\bar{F}(\boldsymbol{Q}) + \bar{S}(\boldsymbol{Q}) + \bar{H}(\boldsymbol{Q}). \tag{27}$$

3.3 Radial Basis Function Partition of Unity Method (RBF-PUM)

The Partition of Unity Method (PUM) was first proposed by Babuška and Melenk [31] and applied the method to solve PDEs. The main idea of RBF-PUM is to subdivide the domain into overlapping subdomains or patches in which we construct the local RBF approximation on each patches, then the global solution is given by the sum of these local approximations multiplied by partition of unity weight functions.

3.3.1 Method's principle

Let $\{\Omega_i\}_{i=1}^{n_p}$ be an open covering of an open set Ω i.e., $\Omega \subseteq \bigcup_{i=1}^{n_p} \Omega_i$. The RBF-PUM is a localized method based on subdividing the domain Ω on np patches $\Omega_1, \ldots, \Omega_{n_p}$. First, we define a partition of unity functions $\{\omega_i\}_{i=1}^{n_p}$ subordinated to the covering $\{\Omega_i\}_{i=1}^{n_p}$ such that

$$\sum_{i=1}^{n_p} \omega_i(x) = 1, \qquad x \in \Omega, \tag{28}$$

where the weight function $\omega_i : \Omega_i \to \mathbb{R}$ is compactly supported, non-negative and continuous. For each patch we may thus construct a local RBF interpolant $I_u^i : \Omega_i \to \mathbb{R}$ of the form:

$$I_u^i(x) = \sum_{j=1}^{n_i} \lambda_j^i \varphi(\|x - x_j^i\|), \tag{29}$$

where n_i is the number of nodes in Ωi. Therefore, the global RBF-PUM interpolant is defined as

$$I_u(x) = \sum_{i=1}^{n_p} \omega_i(x) I_u^i(x) = \sum_{i=1}^{n_p} \omega_i(x) \sum_{j=1}^{n_i} \lambda_j^i \varphi\left(\left\|x - x_j^i\right\|\right), \quad x \in \Omega, \quad (30)$$

By enforcing the interpolation condition, we have the following form of the global interpolant [32]

$$U = \sum_{i=1}^{n_p} R_i W_i A_i \overline{\lambda^i}, \quad (31)$$

where U is a vector containing the global solution atx_k, $k = 1, \dots, N$, R_i is a permutation projection operator which maps the local index set into the global one, W_i is a diagonal matrix with element $\omega_i(x)$ on it and A_i is the local RBF matrix. The weight function ω_i are constructed using Shepard's method [33]

$$\omega_i(x) = \frac{\psi_i(x)}{\sum_{l=1}^{n_p} \psi_l(x)}, \qquad i = 1, \dots, n_p \quad (32)$$

where $\psi(x)$ is compactly supported function with support on Ω_i. We have used the following compactly supported Wendland function [34]

$$\psi(r) = \begin{cases} (1-r)^2(32r^3 + 25r^2 + 8r + 1) & 0 \le r \le 1, \\ 0 & r > 1 \end{cases}$$

In this paper, we will use circular patches due to their flexibility in practice. Thus, the Wendland functions will be scaled to get

$$\psi_i(x) = \psi\left(\frac{\|x - \xi_i\|}{\rho_i}\right), \qquad i = 1, \dots, n_p \quad (33)$$

where ξ_i and ρ_i are, respectively, the centres and the radii of patches Ω_i, $i = 1, \dots, n_p$. Let u_k^i be the value of local solution at the node x_k located in Ω_i. Because our final target is to solve PDEs, the problem here is that there would be more unknown than equations. This can be fixed by requiring the local solutions u_k^i to coincide with the global one. The interpolation property implies that

$$u^i = A_i \overline{\lambda^i} \quad \Rightarrow \quad \overline{\lambda^i} = {A_i}^{-1} u^i \quad (34)$$

Therefore, the approximation of any linear differential operator L can be derived

$$LU = \sum_{i=1}^{n_p} R_i L(W_i A_i) \overline{\lambda^i} \quad (35)$$

$$= \sum_{i=1}^{n_p} R_i L(W_i A_i) {A_i}^{-1} u^i \quad (36)$$

$$= \sum_{i=1}^{n_p} R_i D_L^i u^i \quad (37)$$

where D_L^i is a local differentiation matrix which defined as $D_L^i = L(W_i A_i)A_i{}^{-1}$.

3.2.2 Spatial discretization of SWEs using RBF-PUM

We intend to illustrate the discretization of RBF–PU method for the shallow water equations. Let $\{x_1, x_2, \ldots, x_N\}$ be a set of N distinct points. Using (35), the first and second spatial derivatives of a function $\boldsymbol{Q}(x,t)$ can be expressed as follows:

$$\frac{\partial \boldsymbol{Q}(x,t)}{\partial x} = \sum_{i=1}^{n_p} R_i D_x^i \boldsymbol{Q}^i, \quad \frac{\partial^2 \boldsymbol{Q}(x,t)}{\partial x^2} = \sum_{i=1}^{n_p} R_i D_{xx}^i \boldsymbol{Q}^i, \tag{38}$$

Hence, the global differentiation matrices $\overline{D_x}$ and $\overline{D_{xx}}$ is computed by assembling the above local matrices into the global one, (38) becomes

$$\frac{\partial \boldsymbol{Q}(x,t)}{\partial x} = \overline{D_x}\,\boldsymbol{Q}(x,t), \qquad \frac{\partial^2 \boldsymbol{Q}(x,t)}{\partial x^2} = \overline{D_{xx}}\,\boldsymbol{Q}(x,t), \tag{39}$$

The system (8) is approximated by the following equivalent system

$$\partial_t \boldsymbol{Q} = -\bar{F}(\boldsymbol{Q}) + \bar{S}(\boldsymbol{Q}) + \bar{H}(\boldsymbol{Q}). \tag{40}$$

Using the RBF-PUM to perform numerical evaluation, we obtain

$$\bar{F}(\boldsymbol{Q}) = \begin{pmatrix} \overline{D_x} q \\ \overline{D_x}\left(\frac{q^2}{h} + \frac{1}{2} g h^2\right) \end{pmatrix}, \qquad \bar{S}(\boldsymbol{Q}) = \begin{pmatrix} 0 \\ -gh\overline{D_x} b \end{pmatrix}, \qquad \bar{H}(\boldsymbol{Q}) = \begin{pmatrix} \mu\overline{D_{xx}} h \\ \mu\overline{D_{xx}} q \end{pmatrix}, \tag{41}$$

Equivalent to

$$\partial_t \boldsymbol{Q} = RHS_{RBF-PUM}(\boldsymbol{Q}), \text{ where } RHS_{RBF-PUM}(\boldsymbol{Q}) = -\bar{F}(\boldsymbol{Q}) + \bar{S}(\boldsymbol{Q}) + \bar{H}(\boldsymbol{Q}). \tag{42}$$

3.4 RBF-PUM with QR factorization (RBF-PUM-QR)

It is well known that the use of infinitely smooth RBFs (Table 1) can provide spectral accuracy for solving PDEs. As mentioned previously, this type of RBFs is usually formulated by including a free shape parameter ε, which is generally applied to control the fatness of the functions. Decreasing ε often improves the accuracy of approximation. However, mathematical investigations show that when $\varepsilon \to 0$ (flat limit region), the RBF methods suffers from severe ill-conditioning. Therefore, RBF-PUM requires a stable evaluation method to converge as $\varepsilon \to 0$ [38, 39]. The obvious question becomes how to devise algorithms that balance numerical stability and computational efficiency when dealing with small values of ε. This was the subject of many papers published during the last two decades (e.g. [40, 41, 42]).

In order to use small values of the shape parameter in RBF-PUM, we consider the RBF-QR algorithm [40]. The essential concept behind the RBF-QR algorithm is finding a way to form a well conditioned basis in the same function space spanned by a finite set of nearly flat RBFs [41]. The change of basis will allow to remove the ill-conditioning issue while maintaining the same accuracy or even better one. The construction of the new basis was done by expanding the Gaussian RBF (Table (1)) on Chebyshev polynomials in one dimension, on Chebyshev polynomials and trigonometric functions in two dimensions and on a combination between Chebyshev polynomials and spherical harmonics in three dimensions. The new basis derived from RBF-QR algorithm spans the same approximation space as the Gaussian basis which produce an accurate numerical solutions and bypasses some limitations, including complications related to the selection of ε, ill-conditioning of RBF-PUM matrices. To have more details on the construction of the new basis, we refer the reader to [40].

3.5 Temporal discretization of the numerical methods

To achieve a higher order of accuracy, it is sensible to use a high order time discretization as well. The ODE systems (20), (21), (42) are integrated in time using the fourth order Runge-Kutta scheme. The procedure to advance the solution from the time t^n to the next time t^{n+1} is carried out as

$$k_1 = \Delta t\, RHS(\boldsymbol{Q}_n),$$

$$k_2 = \Delta t\, RHS\left(\boldsymbol{Q}_n + \frac{1}{2}k_1\right),$$

$$k_3 = \Delta t\, RHS\left(\boldsymbol{Q}_n + \frac{1}{2}k_2\right),$$

$$k_4 = \Delta t\, RHS(\boldsymbol{Q}_n + k_3),$$

$$\boldsymbol{Q}_{n+1} = \boldsymbol{Q}_n + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4), \qquad (43)$$

where RHS can be RHS_{kansa} or RHS_{RBF-FD} or$RHS_{RBF-PUM}$, n represents the time level and Δt is the time step, this needs to be chosen carefully to guarantee the stability of the scheme. In all our simulations, we chose Δt using the following formula:

$$\Delta t = CFL\frac{d_{min}}{\max(|u|+\sqrt{gh})}, \qquad (44)$$

where d_{min} denotes the smallest nodal distance between collocation points and CFL is the Courant number such that $0 < CFL < 1$. In the following, Algorithm 1 and Algorithm 2 illustrates the full discretization of the SWEs using RBF-FD method and RBF-PUM-QR method, respectively.

Algorithm 1 Full discretization of 1D SWEs using RBF-FD method

1: Enter the required simulation parameters, i.e $\boldsymbol{N, \mu, n_{loc}, T, nt}$ . ..

2: Construct the RBF-FD differentiation matrices :

a: Specify the computational domain $\boldsymbol{\Omega}$.

b: Generate a set of collocation points (nodes) $\{\boldsymbol{x_i}\}_{\boldsymbol{i=1}}^{\boldsymbol{N}}$ in the domain $\boldsymbol{\Omega}$.

c: Choose an RBF $\boldsymbol{\varphi}$ and determine the shape parameter $\boldsymbol{\varepsilon}$.

d: For each node $\boldsymbol{x_i}$, find a stencil of neighboring nodes $\{\boldsymbol{x_1^i, x_2^i, \dots, x_{n_{loc}}^i}\}$.

e: Form a local RBF interpolation matrix using the stencil nodes.

f: Construct the differentiation matrices $\boldsymbol{W_x}$ and $\boldsymbol{W_{xx}}$ by differentiating the RBFs and solving for the weights the system (22).

3: Make the right hand side of ODE obtained :

$$\boldsymbol{RHS_h = @(h,q) - W_x q + \mu W_{xx} h} \quad (45.a)$$

$$\boldsymbol{RHS_q = @(h,q) - W_x\left(\frac{q^2}{h} + \frac{1}{2}gh^2\right) - g(h.W_x b) + \mu W_{xx} q} \quad (45.b)$$

4: Enter initial condition.

5: for $\boldsymbol{j = 1 : n_t}$ **do**

6: $\boldsymbol{t = j.dt}$

7: $\boldsymbol{k_1 = dt.RHS_h(h,q)}$ $\qquad$ $\boldsymbol{k_1 = dt.RHS_q(h,q)}$

8: $\boldsymbol{k_2 = dt.RHS_h(h + \frac{k_1}{2}, q)}$ $\qquad$ $\boldsymbol{k_2 = dt.RHS_q(h, q + \frac{k_1}{2})}$

9: $\boldsymbol{k_3 = dt.RHS_h(h + \frac{k_2}{2}, q)}$ $\qquad$ $\boldsymbol{k_3 = dt.RHS_q(h, q + \frac{k_2}{2})}$

10: $\boldsymbol{k_4 = dt.RHS_h(h + k_3, q)}$ $\qquad$ $\boldsymbol{k_4 = dt.RHS_q(h, q + k_3)}$

11: $\boldsymbol{h = h + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4)}$ $\qquad$ $\boldsymbol{q = q + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4)}$

12: Apply boundary condition.

13: end

14: Compute error.

Algorithm 1 Full discretization of 1D SWEs using RBF-PUM-QR method

1: Enter the required simulation parameters, i.e $\boldsymbol{N, \mu, n_p, T, nt}$ . ..

2: Construct the RBF-PUM-QR differentiation matrices :

a: Specify the computational domain $\boldsymbol{\Omega}$.

b: Generate a set of collocation points (nodes) $\{\boldsymbol{x_i}\}_{\boldsymbol{i=1}}^{\boldsymbol{N}}$ in the domain $\boldsymbol{\Omega}$.

c: Divide the domain into overlapping subdomains $\{\Omega_i\}_{i=1}^{n_p}$ such that $\boldsymbol{\Omega \subseteq \cup_{i=1}^{n_p} \Omega_i}$.

d: For each subdomain Ω_i, select a set of nodes $\{\boldsymbol{x_j^i}\}_{j=1}^{n_i}$ within Ω_i .

e: For each point $\{\boldsymbol{x_j^i}\}_{j=1}^{n_i}$, define the partition of unity using Shepard's method (32).

f: Construct the differentiation matrices $\overline{\boldsymbol{D_x}}$ and $\overline{\boldsymbol{D_{xx}}}$ using the differentiation weights obtained from the QR factorization of the local RBF-PUM systems, specifically using Gaussian RBFs [40].

3: Make the right hand side of ODE obtained :

$$\boldsymbol{RHS_h = @(h,q) - \overline{D_x}\, q + \mu \overline{D_{xx}}\, h} \quad (46.a)$$

$$\boldsymbol{RHS_q = @(h,q) - \overline{D_x}\left(\frac{q^2}{h} + \frac{1}{2} g h^2\right) - g(h.\overline{D_x}\, b) + \mu \overline{D_{xx}}\, q} \quad (46.b)$$

4: Enter initial condition.
5: for $\boldsymbol{j = 1 : n_t}$ **do**
6: $\boldsymbol{t = j.dt}$
7: $\boldsymbol{k_1 = dt.RHS_h(h,q)}$ $\boldsymbol{k_1 = dt.RHS_q(h,q)}$
8: $\boldsymbol{k_2 = dt.RHS_h(h + \frac{k_1}{2}, q)}$ $\boldsymbol{k_2 = dt.RHS_q(h, q + \frac{k_1}{2})}$
9: $\boldsymbol{k_3 = dt.RHS_h(h + \frac{k_2}{2}, q)}$ $\boldsymbol{k_3 = dt.RHS_q(h, q + \frac{k_2}{2})}$
10: $\boldsymbol{k_4 = dt.RHS_h(h + k_3, q)}$ $\boldsymbol{k_4 = dt.RHS_q(h, q + k_3)}$
11: $\boldsymbol{h = h + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4)}$ $\boldsymbol{q = q + \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4)}$
12: Apply boundary condition.
13: end
14: Compute error.

3.6 Friction source term discretization

The non-linear nature of the friction terms (2), (3) and their interactions with other source terms remains to be a challenge for developing numerically accurate and stable schemes to solve the SWEs for simulating very shallow flows as found in the applications involving overland flows and wet/dry fronts. Indeed, the friction term actually dominates the stability of a numerical scheme. In particular, when the water depth becomes very small, the friction formulations may lead to an exaggerated force that can even reverse the flow, which is obviously physically incorrect [43]. To deal with this problem, at the beginning of each step of the Runge-Kutta scheme, the friction effect is evaluated and used implicitly by the splitting method described by [27, 43], and it is equivalent to solve the following ordinary differential equations

$$\frac{\partial q}{\partial t} = S_f. \quad (47)$$

Since the friction term is only involved in the momentum equation, only the flow rates need to be evaluated. The above equation is then discretized by a full implicit method as:

$$\frac{q^{n+1} - q^n}{\Delta t} = S_f^{n+1}, \quad (48)$$

where the friction term S_f^{n+1} may be expressed using a Taylor series as

$$S_f^{n+1} = S_f^n + \left(\frac{\partial S_f}{\partial q}\right)^n \Delta q + o(\Delta q^2), \quad (49)$$

Where $\Delta q = q^{n+1} - q^n$. Ignoring the higher-order terms and substituting the above equation into (48) lead to the following formula for updating water discharge q at the new time step:

$$q^{n+1} = q^n + \Delta t \frac{S_f^n}{1-\Delta t\left(\frac{\partial S_f}{\partial q}\right)^n} \quad (50)$$

$$= q^n + \Delta t \left(\frac{S_f}{D}\right)^n = q^n + \Delta t F, \quad (51)$$

where

$$D = \begin{cases} 1 + 2\Delta t \left(\frac{gn^2|u|}{h^{\frac{4}{3}}}\right) \\ 1 + 2\Delta t \left(\frac{f|u|}{8h}\right) \end{cases} \quad (52)$$

D is the coefficient derived for a full implicit scheme where the first line is for manning law and the second line is for Darcy-Weisbach law and F is the friction source term including the implicit coefficient. This updated water discharge is used as an initial condition for the operators in equation (43).

4. Numerical results

In order to verify the feasibility and the ability of the proposed methods combined with artificial viscosity to solve the problem with strong discontinuity appears in SWEs, we present sets of numerical tests, including various frictionless steady-state solutions, transient solutions and steady-state solutions with friction to address different numerical challenges. These problems are widely used to test numerical algorithms for the SWEs (e.g. [44, 45, 51]). The goal of the first numerical test is to assess the ability of the suggested schemes to capture steady state solutions, in addition to preserving them. The second numerical example is performed to test if the numerical methods catches the transitory behaviour and shock behaviour of the solution properly. The third numerical test verifies the accuracy of the proposed methods in preserving moving steady states that involve both topography and friction. The following parameters are defined: N is the total number of nodes in the whole computational domain, n_{loc} is the stencil number of the RBF-FD method based on the inverse multiquadric radial basis function, the optimal shape parameter ε is selected using the computed error of the approximate solutions, n_p is the number of patch of the RBF-PUM-QR method. For Kansa method based on the multiquadric radial basis function, the shape parameter $\varepsilon = \frac{\varepsilon_0\sqrt{N}}{d_{min}}$ in which d_{min} is the minimum distance between two nodes and $10^{-4} \leq \varepsilon_0 \leq 10^{-1}$ [46] and T is the terminal time of the simulation. To show the advantages of the methods in terms of accuracy, we compute the relative error as follows:

$$Err = \sqrt{\frac{1}{N}\sum_{i=1}^{N}\left(\frac{\boldsymbol{Q}^{num}(x_i,t)-\boldsymbol{Q}^{ana}(x_i,t)}{\boldsymbol{Q}^{ana}(x_i,t)}\right)^2}, \quad (53)$$

where $\boldsymbol{Q}^{num}$ and $\boldsymbol{Q}^{ana}$ are respectively, the numerical and analytical solutions. In all our computations a fixed courant number $CFL = 0.5$. The numerical tests are performed on a core CPU i5 2.11GH computer in a MATLAB 2018a tools.

4.1 Test 1: Steady flow over a bump

According to [47], the source term appears in (8) is a crucial point in preserving steady states. In order to prove the ability of the proposed methods to catch these states, we apply the schemes to a series of benchmark cases of different boundary and initial conditions [48]. These benchmarks have been derived by considering the Bernoulli equation that governs the steady state solutions of the shallow-water equations with non-flat topography and a vanishing friction contribution, The experiments from [48] are called the *subcritical flow*, the *transcritical flow without shock* and the *transcritical flow with shock*. The computational domain is a rectangular channel with $L = 25\,m$ in length. The inflow and outflow boundary conditions are imposed along the left and the right boundary segments, respectively. The bed elevation of a bump is defined as follows:

$$b(x) = \begin{cases} 0.2 - 0.05(x-10)^2 & 8m < x < 12m \\ 0 & otherwise. \end{cases} \quad (54)$$

The boundary conditions are defined in terms of two quantities, q_0 and h_L, whose values vary depending on the specific experiment being analysed:

- on the left boundary, the water height satisfies a homogeneous Neumann condition and the discharge is set to a specific q_0.
- on the right boundary, the water height is set to h_L when the flow is subcritical (and a homogeneous Neumann boundary condition is prescribed otherwise), and the discharge follows a homogeneous Neumann boundary condition.

In addition, the initial conditions set to $h(x) + b(x) = h_L m$ and $q(x) = 0m^2/s$ throughout the domain.
All these results are frictionless and displayed at $T = 200s$ using$N = 1001$, $n_{loc} = 100$ and $n_p = 300$, the analytical solutions are also plotted within the obtained numerical results.

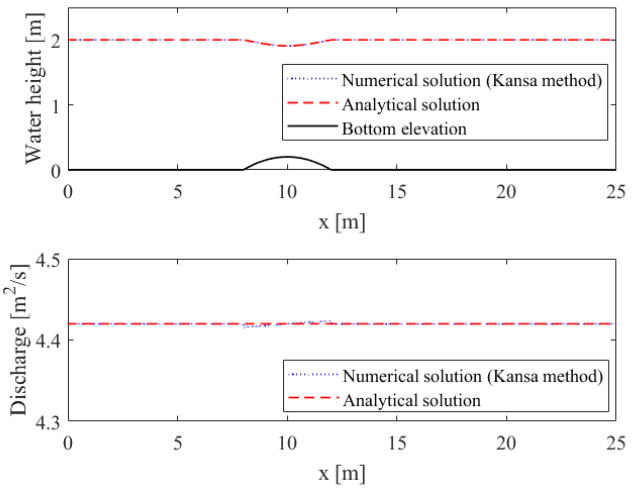
4.1.1 Subcritical flow

For this test, a discharge of $q_0 = 4.42m^2/s$ and the constant water surface level of the water level $h_L = 2m$ are set as upstream and downstream boundary conditions, respectively. The steady-state numerical solutions of water height h and discharge q compared with the analytical solution [48] are plotted in Figure 1. The numerical examples are performed for Kansa method using $\mu = 8 \times 10^{-3}$, for

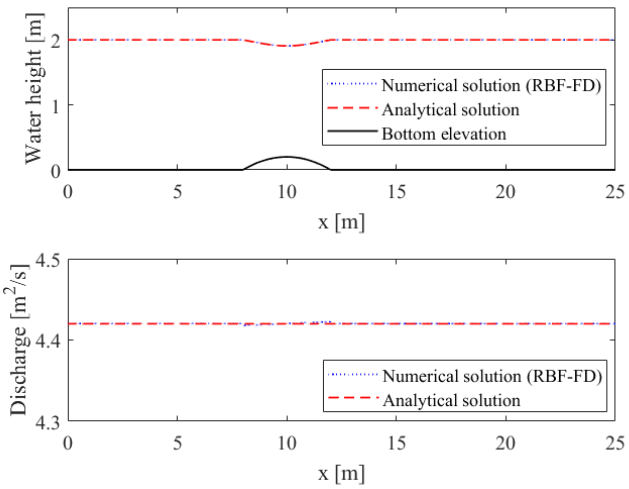
RBF-PUM-QR method using $\mu = 8 \times 10^{-3}$, and for RBF-FD method using $\mu = 2 \times 10^{-3}$ with the optimal shape parameter $\varepsilon = 5$. The selection of an optimal shape parameter ε and the viscosity coefficient μ in all tests in this paper was made using trial and error strategy. The strategy is to perform a series of experiments by varying the values of the shape parameter or viscosity coefficient, and then pick the best one corresponding to the smallest error. In Figure 1, it is evident that the impact of a bump in subcritical flow is clearly observable and the model can well capture the drop of the water surface elevation at the bump. A more detailed analysis of the numerical error is listed in Table 2. Figure 2 shows the computed L^2 error for different values of the viscosity coefficient μ. From Figure 2, we remark that for Kansa method the minimum error is for obtained around $\mu = 10^{-3}$, for RBF-FD method $\mu = 10^{-2}$ and for RBF-PUM-QR method $\mu = 10^{-3}$. We make a error analysis for the subcritical flow problem using RBF-FD method (see Figure 3 (a)). Four different stencil sizes are used, $n_{loc} = 20$, $n_{loc} = 100$ and $n_{loc} = 150$. The result seen in Figure 3 (a) plotted in logarithmic scales , where the L^2-norm of the numerical error is plotted against the grid size, we display in Figure 3 (b) the error of the numerical solution at $T = 200$ seconds in terms of the shape parameter in order to show that stable computations were achieved for $\varepsilon = 5$. The computed solutions are oscillation-free which demonstrates the ability of the proposed methods to accurately capture steady state solutions.

Table 2. Error comparison of water level and discharge for different flow conditions (*subcritical, transcritical without shock and transcritical with shock*) using Kansa, RBF-FD and RBF-PUM-QR methods.

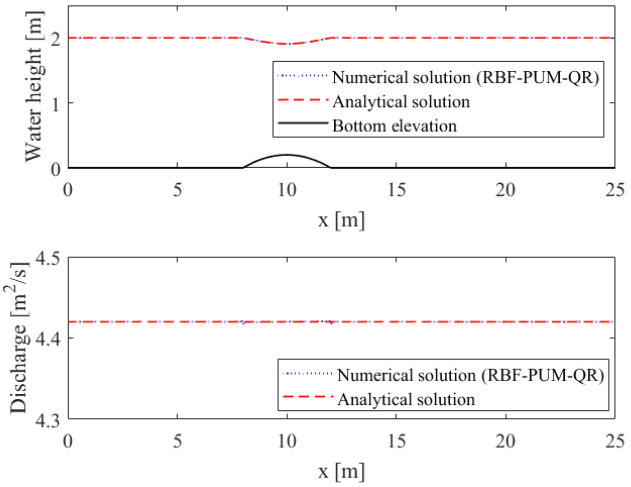
		Kansa	RBF-FD	RBF-PUM-QR
Subcritical	Water height $h[m]$	6.73×10^{-6}	3.94×10^{-6}	1.38×10^{-6}
	Discharge $q[m^2/s]$	7.02×10^{-6}	4.21×10^{-6}	1.05×10^{-6}
Transcritical without shock	Water height $h[m]$	3.99×10^{-5}	1.95×10^{-5}	1.26×10^{-5}
	Discharge $q[m^2/s]$	1.88×10^{-5}	1.06×10^{-5}	6.90×10^{-6}
Transcritical with shock	Water height $h[m]$	4.70×10^{-4}	4.51×10^{-4}	4.40×10^{-4}
	Discharge $q[m^2/s]$	3.50×10^{-4}	2.50×10^{-4}	3.10×10^{-4}



(a) Kansa method



(b) RBF-FD method



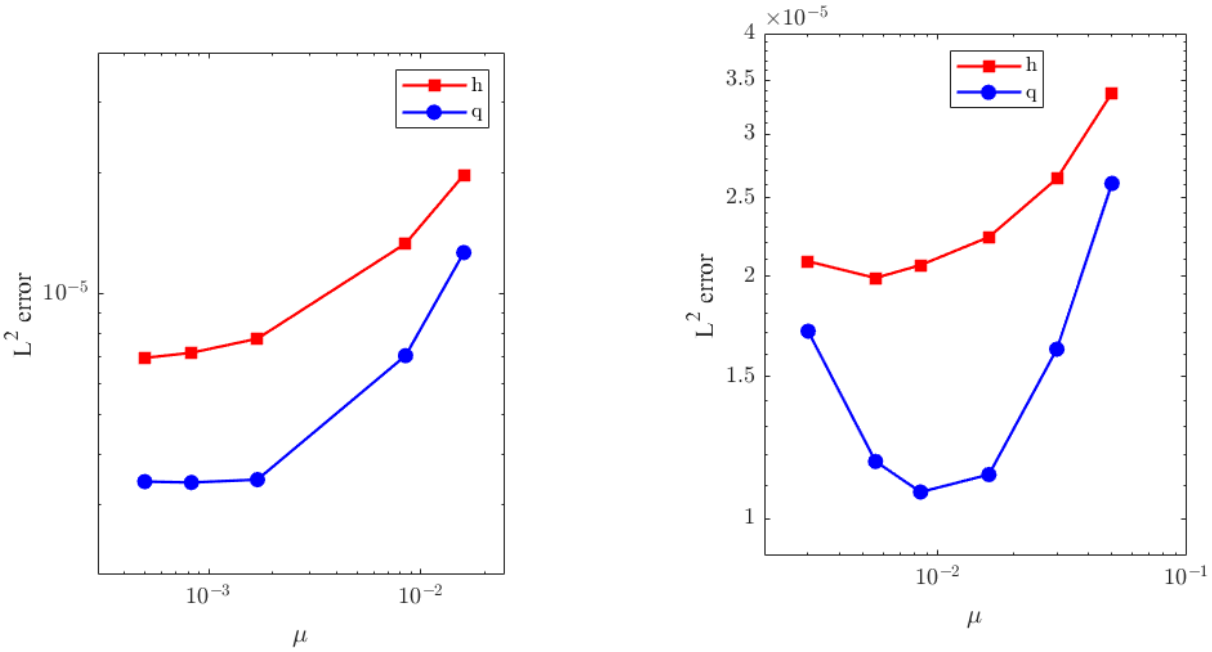
(c) RBF-PUM-QR method

Figure 1. Subcritical flow over a bump using (a) Kansa, (b) RBF-FD, and (c) RBF-PUM-QR at T = 200 seconds

4.1.2 Transcritical flow without shock

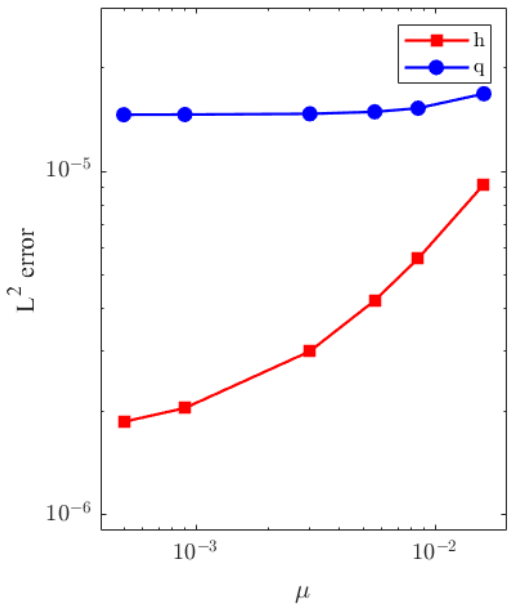
In this case, the upstream inflow of the discharge $q_0 = 1.53m^2/s$ is imposed and a downstream condition for the height $h_L = 0.40m$ only if the flow is subcritical. If the flow is supercritical, no condition is imposed. The numerical examples are performed for Kansa method using $\mu = 7 \times 10^{-3}$, for RBF-FD method using $\mu = 4 \times 10^{-3}$, the optimal shape parameter $\varepsilon = 7$ and for RBF-PUM-QR method using $\mu = 10^{-3}$. As shown in Figure 4, the water height and discharge are compared with the analytical solution [48], which can be observed that the water surface drops significantly as the flow passes through the bump. The transcritical flow happened nearby the bump, since the subcritical flow is at the upstream and the supercritical flow is at the downstream. The errors of water height h and

discharge q are listed in Table 2. It can be clearly seen that the numerical and analytical solutions are in good agreement.



(a) Kansa method

(b) RBF-FD method



(c) RBF-PUM-QR method

Figure 2. L^2 error as a function of the viscosity coefficient for the ”Subcritical flow” problem for (a) Kansa, (b) RBF-FD, and (c) RBF-PUM-QR at T = 200 seconds

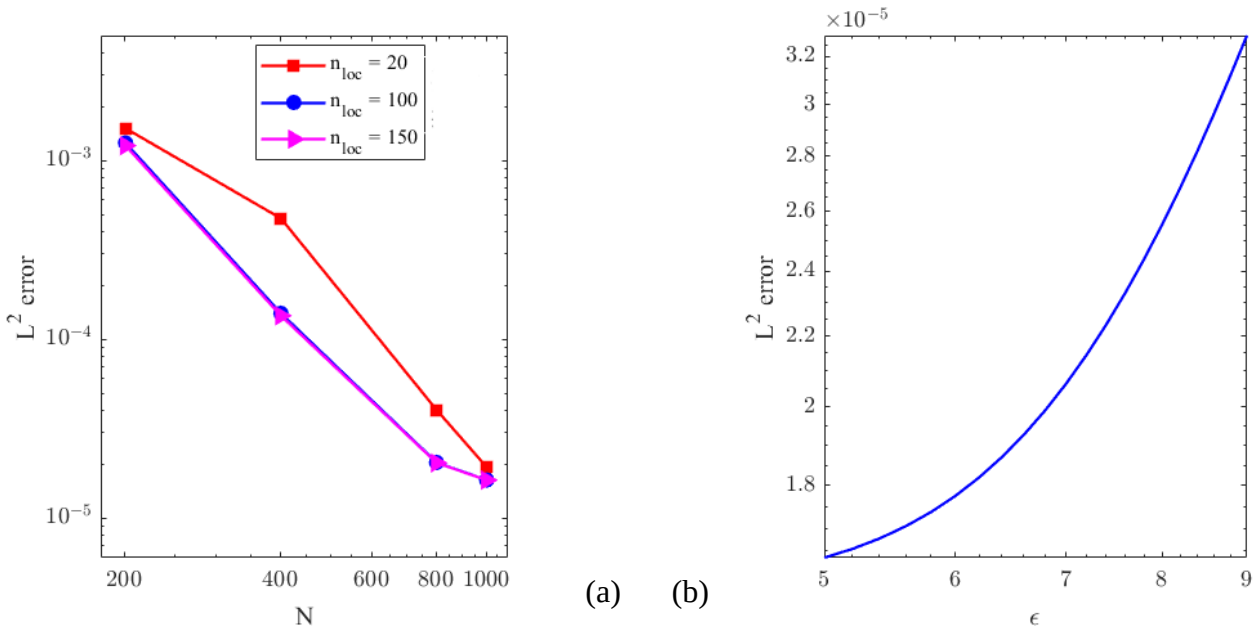


Figure 3.

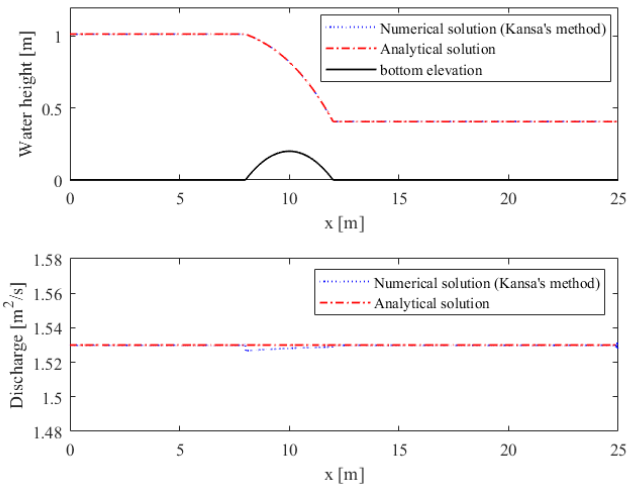
(a) L^2 Errors of the RBF-FD method for solving "Subcritical flow" problem using $\mu = 8 \times 10^{-3}$. (b) L^2 error in terms of the shape parameter ε.

4.1.3 Transcritical flow with shock

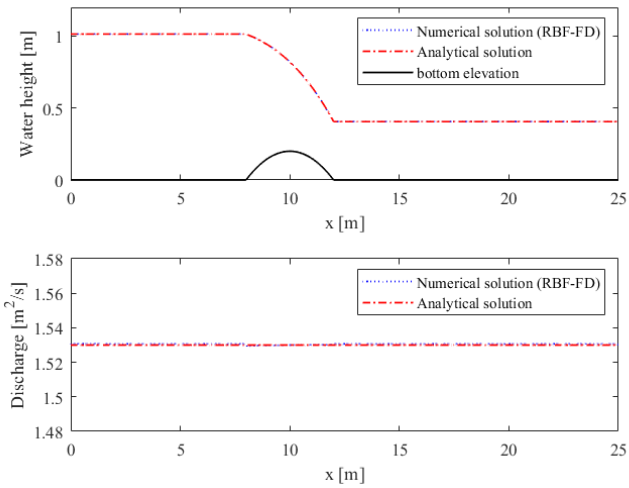
This case is similar to the previous one but with different boundary conditions. Here, a discharge of $q_0 = 0.18m^2/s$ is imposed as the upstream boundary condition and a water level of $h_L = 0.33m$ as the downstream boundary condition; The flow regime changed from subcritical flow to supercritical flow and back to subcritical flow with a hydraulic jump over the bump. For this numerical test, we used the following parameters $\mu = 8 \times 10^{-3}$, $\mu = 6 \times 10^{-3}$ and $\mu = 9 \times 10^{-3}$, for Kansa, RBF-FD, RBF-PUM-QR methods respectively. In our simulation, we set the shape parameter $\varepsilon = 5$. Figure 5, shows the numerical solutions concerning the water height and discharge, compared with the analytical solution. However, near the jump discontinuity we can observe differences that are even more clear when we compare analytical and numerical discharge. A more quantitative analysis of the accuracy is listed in Table 1, we observe that capturing the discharge q correctly in the Kansa method is more difficult than in the other methods. From the results and comparisons in this example, the merits of the proposed meshless methods are verified.

4.2 Test 2: Dam break problem

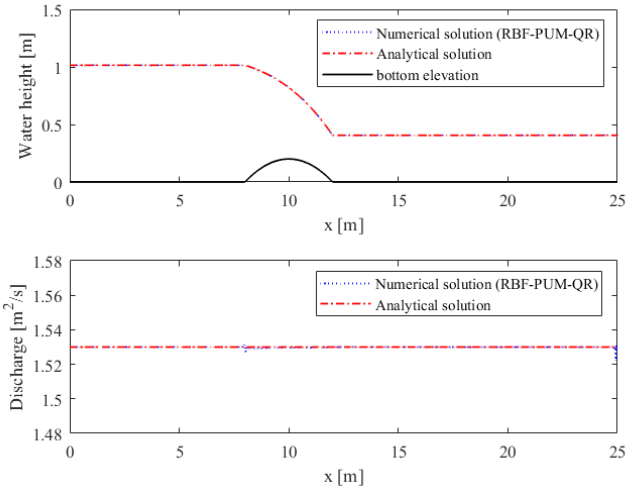
The dam-break problem is the most commonly investigated problem and have also become the standard test scenario for numerical methods for shallow water equations. Toro [49] has the dam-break



(a) Kansa method



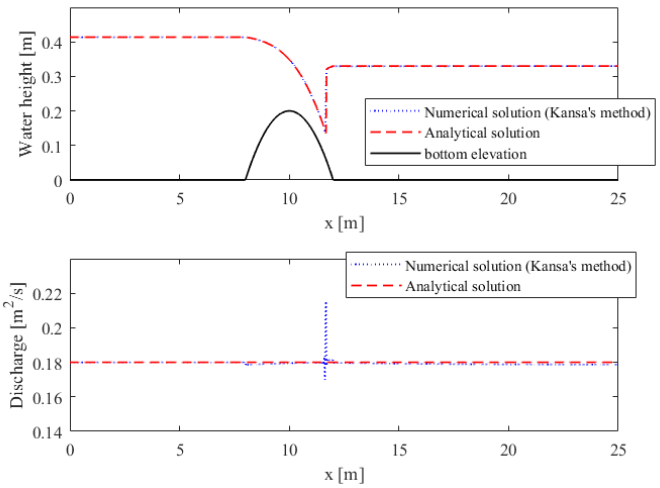
(b) RBF-FD method



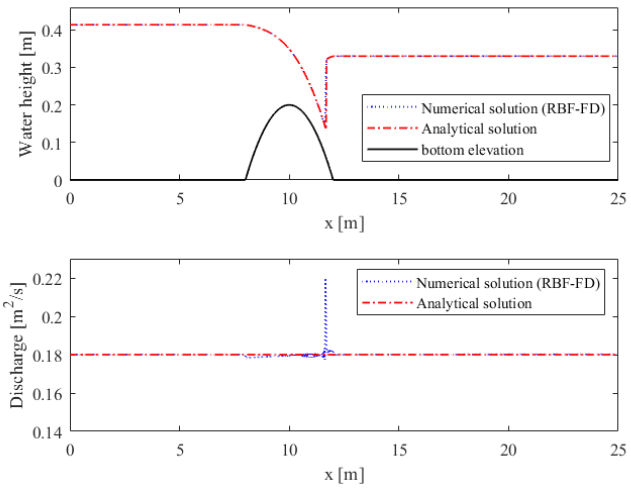
(c) RBF-PUM-QR method

Figure 4. Transcritical flow without shock over a bump using (a) Kansa, (b) RBF-FD, and (c) RBF-PUM-QR at T = 200 seconds

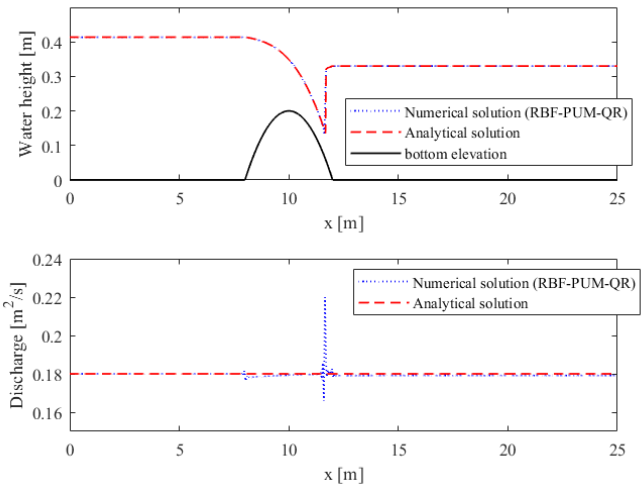
in high regard, because its solution includes both continuous and discontinuous solutions at the same time, and several researches have shown that the shallow water equations are suitable for the representation of dam-break flows e.g. [3, 50, 51] and thus, in order to examine the capability of the proposed methods considering artificial viscosity term to solve the problem of discontinuous initial condition, we consider dam-break test is adopted as the second test. The one-dimensional dam break is an initial value problem consisting of two still bodies of water of different heights. The upstream body (the reservoir) is separated from the downstream body of water by a partition that is instantaneously removed at time $t = 0$ seconds. The two bodies of water are allowed to interact under the force of gravity. We consider the dam-break problem in a rectangular channel with flat bottom, $b(x) = 0$ and frictionless. The channel is of length $10m$ and the initial conditions are given by



(a) Kansa method



(b) RBF-FD method



(c) RBF-PUM-QR method

Figure 5. Transcritical flow with shock over a bump using (a) Kansa, (b) RBF-FD, and (c) RBF-PUM-QR at T = 200 seconds

$$q(x,0) = 0 \qquad \text{and} \qquad h(x,0) = \begin{cases} 0.005 & x < 5m \\ 0.001 & otherwise \end{cases} \qquad (55)$$

As boundary conditions, a zero discharge and a free boundary are considered at the left and right ends of the channel. The analytical solution for this simple dam break test consists of a backward propagating rarefaction and a forward-moving shock wave [49]. The comparison of the numerical results at $T = 2, 4, 6\ s$ are displayed in Figure 6 for RBF-FD method using $\mu = 2 \times 10^{-4}$ and the optimal shape parameter $\varepsilon = 12$. Table 3 depicts the error comparison of water height h and discharg

Table 2. Error comparison of water level and discharge for Dam break problem over a wet bed.

	Kansa	RBF-FD	RBF-PUM-QR
Water height $h[m]$	3.07×10^{-4}	1.71×10^{-4}	2.10×10^{-4}
Discharge $q[m^2/s]$	1.48×10^{-3}	9.76×10^{-4}	8.85×10^{-4}

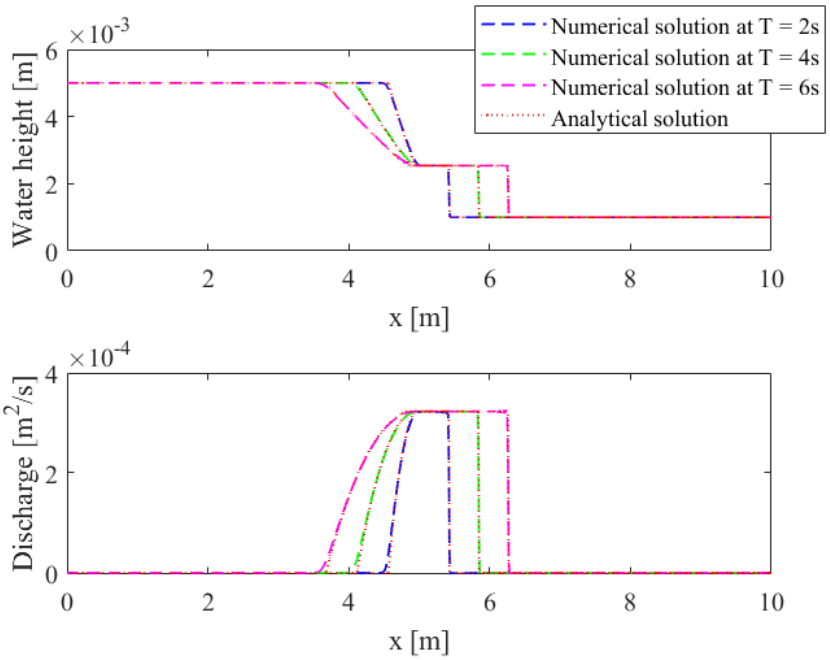
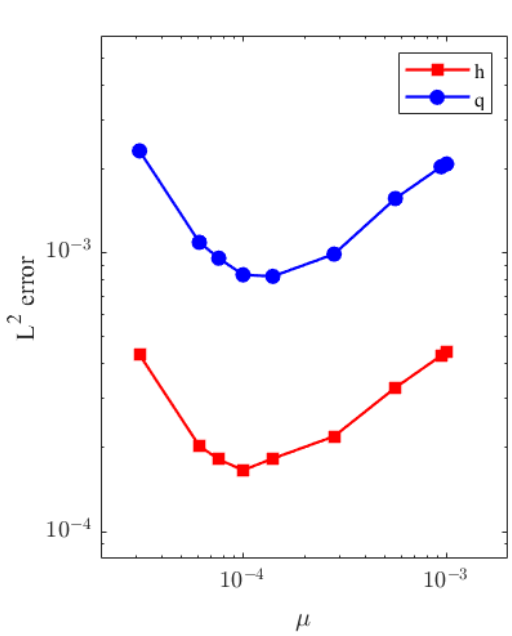


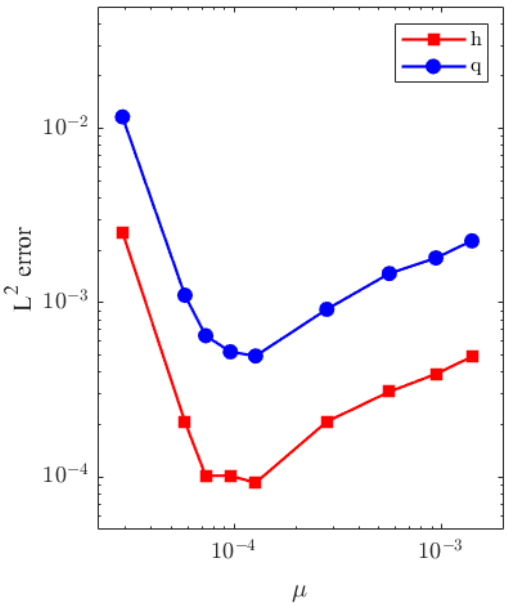
Figure 6: Dam-break on wet bed using RBF-FD method at $T = 2, 4, 6\ s$, top: water height, bottom: discharge.

for all RBF-based methods. The results show that the proposed meshless numerical methods with hyperviscosity has ability of shock capturing and provides stability in dealing with problems with discontinuous flow. Additionally, Figure 7 shows the computed L^2 error for different values of the viscosity coefficient μ, this coefficient controls the accuracy of the method where small values are expected to provide accurate solutions, and should be a limit value of around 10^{-4} to avoid instability issues. Another consequence is that the accuracy of RBF-FD method does not indefinitely increase with stencil size n_{loc}. As shown in Figure 8 (left) the stencil size has no bearing on accuracy, hence $n_{loc} = 100$ is the stencil size chosen for this test case. With $\mu = 10^{-4}$, we display in Figure 8 (right) the curve of the L^2 in terms of the shape parameter ε, we remark that the minimum error is obtained for $\varepsilon = 12$. From this study, we can note that increasing the number of points per stencil leads to enhanced accuracy.

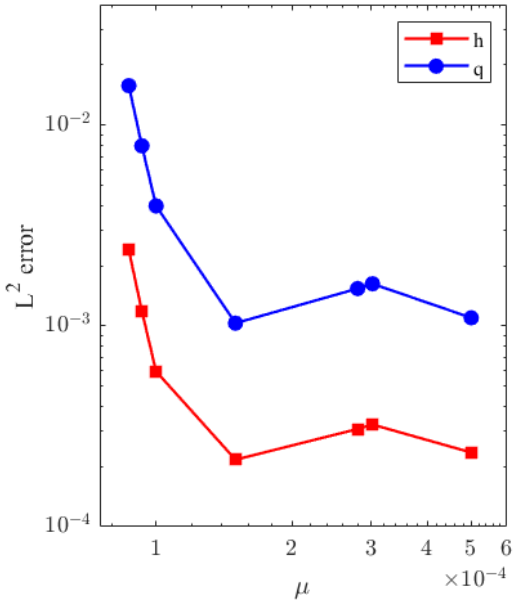
4.3 Test 3: Steady flow with friction



(a) Kansa method

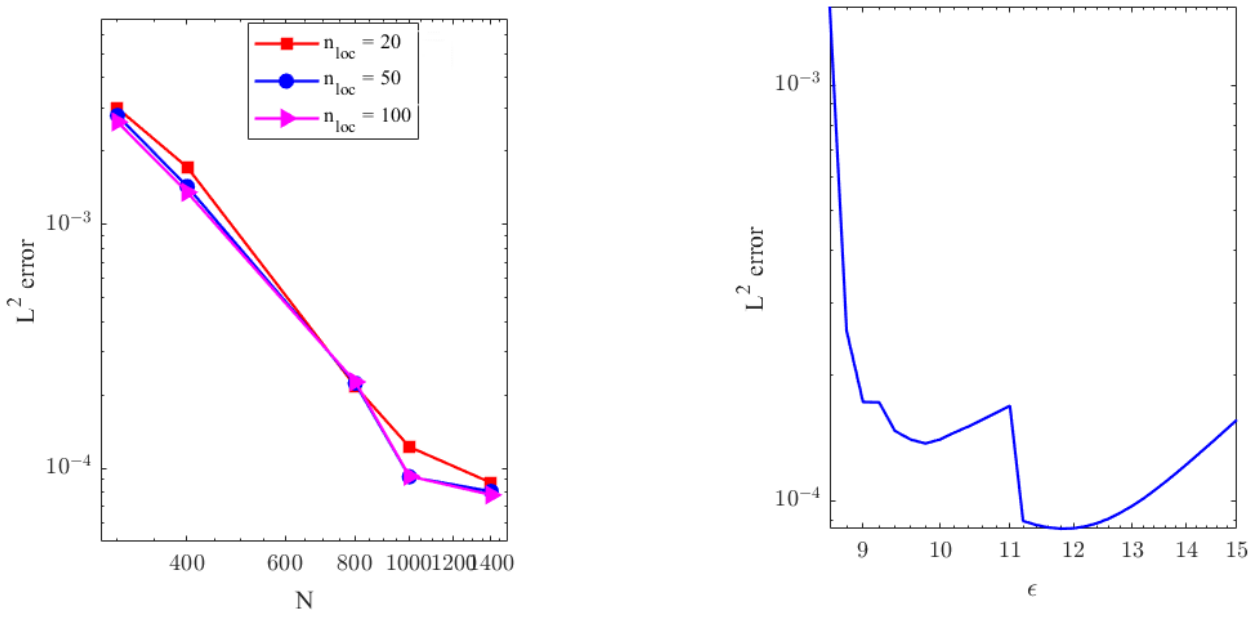


(b) RBF-FD method



(c) RBF-PUM-QR method

Figure 7. L^2 error as a function of the viscosity coefficient for the ”Dam-break” problem for (a) Kansa, (b) RBF-FD, and (c) RBF-PUM-QR at T = 6 seconds



(a) (b)

Figure 8. (a) L^2 Errors of the RBF-FD method for solving "Dam-break" problem with respect to the number of stencils using $\mu = 2 \times 10^{-4}$. (b) L^2 error in terms of the shape parameter ε.

The solutions presented in this section are more intricate than those in the previous section 4.1-4.2 due to the variability of the topography near the boundaries. Consequently, they provide a more rigorous validation of the boundary conditions. When $S_f = 0$ (indicating bottom friction), the following solutions can verify if the friction terms are implemented correctly to maintain steady states. These solutions are derived using the procedure introduced by I. MacDonald [52, 53]: given the water depth profile and the discharge, the corresponding topography is subsequently computed. The three selected solutions illustrate the efficacy of the scheme in addressing stationary states induced by the topography and by friction under a wide range of flow conditions.
At steady states, we have $\partial_t h = \partial_t u = \partial_t q = 0$, thus the mass-conservation equation gives $q =$ constant and we get the equation

$$\partial_x b = \left(\frac{q^2}{gh^3} - 1\right) \partial_x h + S_f. \qquad (56)$$

Where S_f depends on the friction law chosen. From this relation, one can make as many solutions as needed. In this section, we present a few of these solutions obtained for specific value of the domain length L and fixed parameters, such as the friction law and its coefficient. The water height profile and the discharge are provided, and the corresponding topographies are computed by solving the equation (56) with a high order iterative method [51].
In the following computations, the initial conditions set to $h(x) + b(x) = 0, q(x) = 0$ and all results are displayed at $T = 1500s$ using $N = 801, n_{loc} = 100$ and $n_p = 400$. The analytical solutions shown in [51] are also plotted within the obtained numerical results.

4.3.1 Subcritical flow

We consider a $L = 1000\,m$ long channel with a discharge of $q = 2m^2/s$. The flow is subcritical at inflow and is subcritical at outflow with depth $h_{ana}(1000) = 0.748409\,m$ where h_{ana} is analytical solution. The Manning coefficient for the channel is $n = 0.033m^{-1/3}s$. The boundary conditions set to

$$\begin{cases} q(x) = 2 & x = 0 \\ h(x) = h_{ana}(1000) & x = L \end{cases} \tag{57}$$

The water depth and discharge are shown in Figure 9 for Kansa, RBF-FD with an optimal shape parameter ε = 0.25 and RBF-PUM-QR methods using $\mu = 10^{-1}$ in all cases. The numerical solutions is very close to the analytical solution, and we obtained very satisfying results in the approximation of the steady discharge and water height as illustrated in Table 4. Figure 10 shows the computed L^2 error for different values of the viscosity coefficient μ. From Figure 10, we remark that for Kansa method the minimum error is for obtained around $\mu = 10^{-1}$, for RBF-FD method $\mu = 10^{-1}$ and for RBF-PUM-QR method $\mu = 10^{-1}$. We make an error analysis for the subcritical flow problem using RBF-FD method (see Figure 11 (a)). Four different stencil sizes are used, $n_{loc} = 20$, $n_{loc} = 100$ and $n_{loc} = 150$. The result in seen in Figure 11 (a) plotted in logarithmic scales, where the L^2-norm of the numerical error is plotted against the grid size, we display in Figure 11 (b) the error of the numerical solution at $T = 1500$ seconds in terms of the shape parameter in order to show that stable computations were achieved for $\varepsilon = 0.25$. From the results and comparisons in this example, the merits of the proposed meshless method are verified.

4.3.2 Supercritical flow

Considering a $1000\,m$ long computational domain with a discharge of $q = 2m^2/s$. The flow is supercritical at inflow with depth $h_{ana}(0) = 0.741599\,m$ and is supercritical at outflow. The Manning coefficient for the channel is $n = 0.0218m^{-1/3}s$. The boundary conditions set to

$$\begin{cases} q(x) = 2, \quad h(x) = h_{ana}(0) & x = 0, \\ free & x = L. \end{cases} \tag{58}$$

The initial free surface is set to zero on the whole domain. We show on Figure 12 the steady state water free surface and discharge at $T = 1500s$ and we observe very close agreement between numerical results and the reference solution. The numerical examples are performed for Kansa, RBF-FD with an optimal shape parameter $\varepsilon = 0.1$ and RBF-PUM-QR methods using $\mu = 10^{-1}$ for all cases. The error of converged results by different schemes are listed in Table 4.

4.3.3 Supercritical to subcritical flow

We consider a discharge of $\mathrm{q} = 2m^2/s$. Again, the initial free surface and discharge are set to zero on the whole domain. The flow is supercritical at inflow with depth $h_{ana}(0) = 0.543853m$ and is

Table 4. Error comparison of water level and discharge for different flow conditions (*subcritical, supercritical and supercritical to subcritical*) with friction using Kansa, RBF-FD and RBF-PUM-QR methods.

		Kansa	RBF-FD	RBF-PUM-QR
Subcritical	Water height $h[m]$	2.66×10^{-4}	2.29×10^{-4}	2.55×10^{-4}
	Discharge $q[m^2/s]$	1.85×10^{-4}	1.39×10^{-4}	1.82×10^{-4}
Supercritical	Water height $h[m]$	8.88×10^{-5}	1.07×10^{-5}	8.65×10^{-5}
	Discharge $q[m^2/s]$	9.55×10^{-5}	7.83×10^{-5}	6.04×10^{-6}
Supercritical to subcritical	Water height $h[m]$	6.45×10^{-4}	8.31×10^{-4}	5.20×10^{-4}
	Discharge $q[m^2/s]$	2.83×10^{-4}	3.21×10^{-4}	2.50×10^{-4}

subcritical at outflow with depth $h_{ana}(1000) = 1.334899\mathrm{m}$. The Manning coefficient for the channel is $n = 0.0218m^{-1/3}s$. The boundary conditions set to

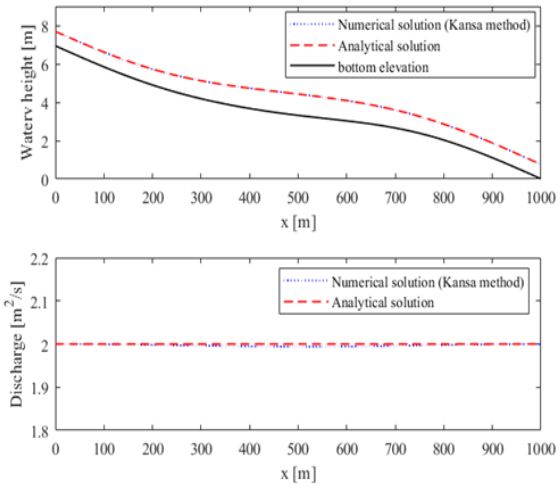
$$\begin{cases} q(x) = 2, \quad h(x) = h_{ana}(0) & x = 0, \\ h(x) = h_{ana}(1000) & x = L. \end{cases} \tag{59}$$

The water depth and discharge are shown in Figure 13 for Kansa, RBF-FD $\varepsilon = 0.1$ and RBF-PUM-QR methods using $\mu = 10^{-1}$. We can clearly see that the flow accurately converges towards the steady state and that the shock location is accurately computed. The comparison of the relative error of the results by Kansa, RBF-FD and RBF-PUM-QR methods is shown in Table 4.

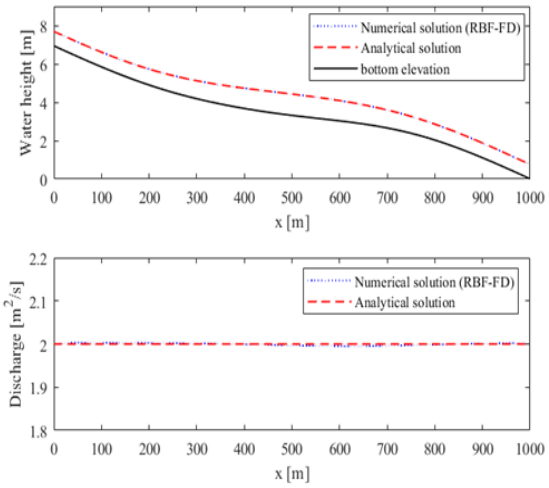
5. Conclusion

In this paper, the shallow water equations is presented to simulate the one-dimensional flow of water in channels. The resulting system is solved numerically using a globally, localized and stable meshless methods based on radial basis functions. In order to stabilize these methods and minimize the non-physical numerical oscillations near the discontinuities which can solve the strong discontinuity

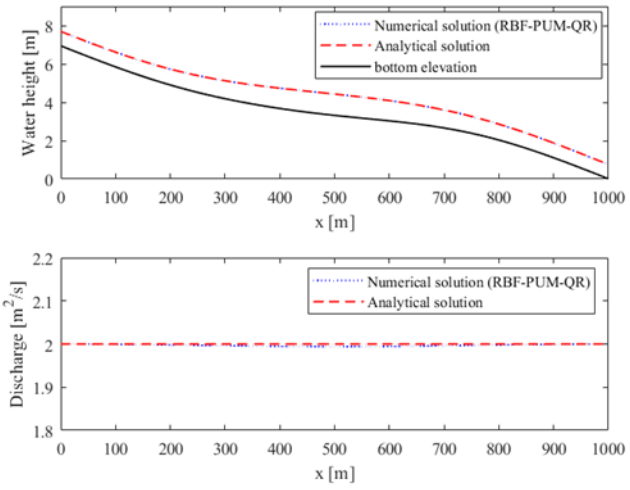
problem in SWEs, an artificial viscosity (Hyperviscosity) technique is implemented. In Kansa's method the system to be solved is dense and therefore it was necessary to refer to localized meshless methods such as RBF-FD and RBF-PUM in order to present sparsity and locality. The problem of choosing any value of shape parameter ε from the flat limit region which affect the accuracy of the solution can be handled by combining the RBF-PUM with RBF-QR approach.
The proposed meshless methods with hyperviscosity has been tested on systems of shallow water equations at different flow regimes. The obtained results indicate good shock resolution with high



(a) Kansa method



(b) RBF-FD method



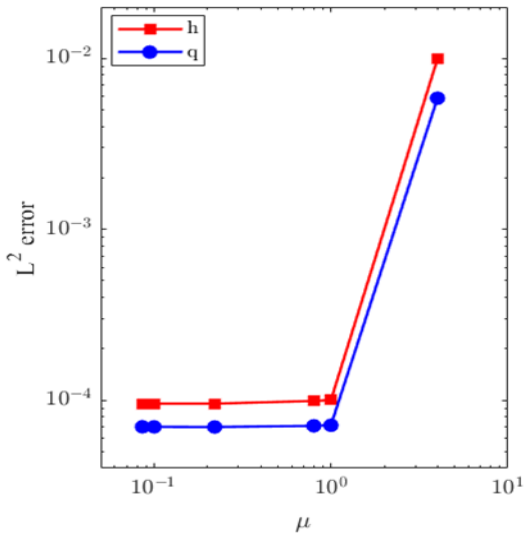
(c) RBF-PUM-QR method

Figure 9. Subcritical flow using (a) Kansa, (b) RBF-FD, and (c) RBF-PUM-QR methods at T = 1500 seconds with Manning friction $n = 0.033m^{-1/3}s$.

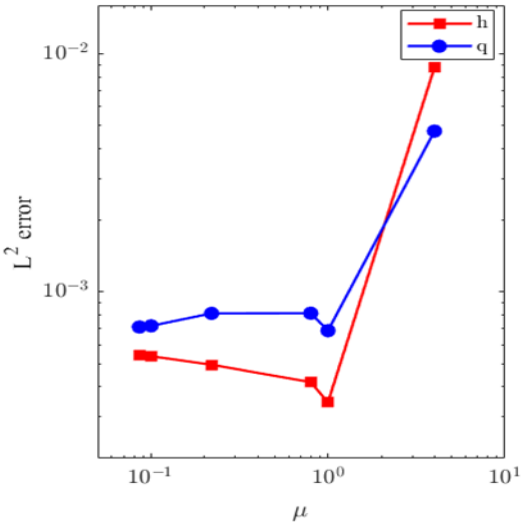
accuracy in smooth regions, and the convergence to the correct steady state solution has been clearly verified in flow over a non-flat bottom, which confirm the numerical stability and computational

accuracy. Additionally, the main challenge encountered was the selection of hyperviscosity parameters, particularly the constant μ and its scaling relations, to ensure adequate stabilization across a wide range of flow regimes and discretization densities.

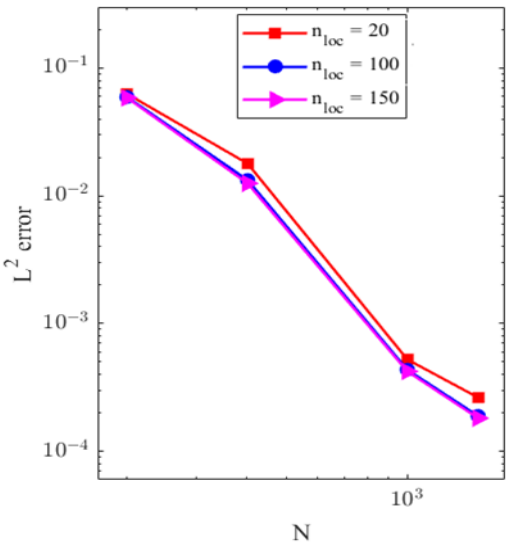
While our numerical computations have been limited to one-dimensional shallow water problems, the current meshless-based RBF methods can be readily extended to two-dimensional shallow water problems incorporating source terms. These extensions, along with other related issues, will be the focus of future investigations.



(a) Kansa method

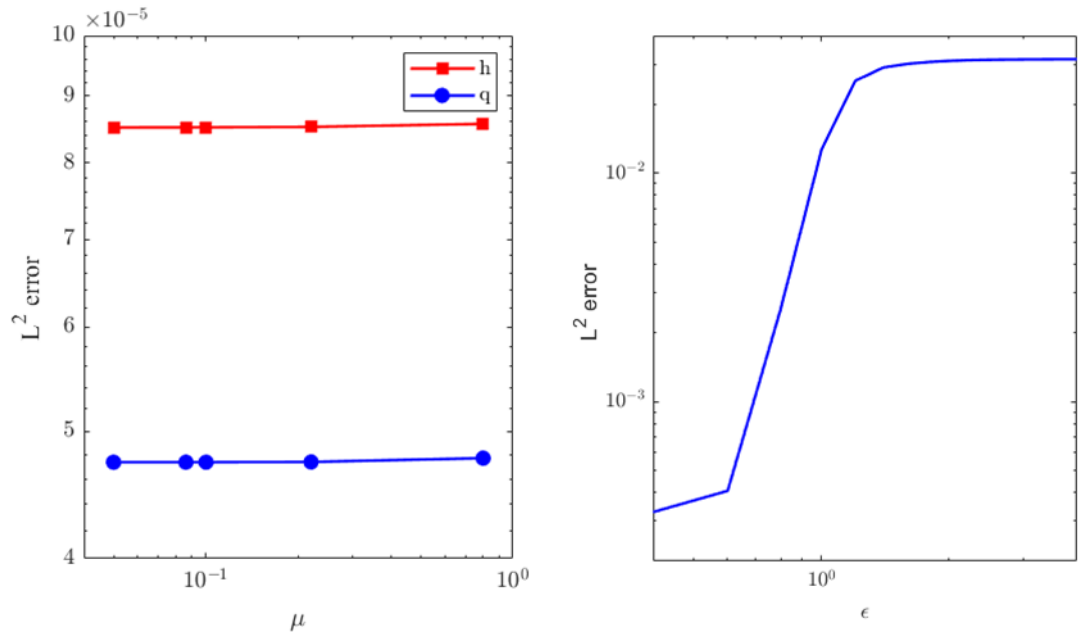


(b) RBF-FD method



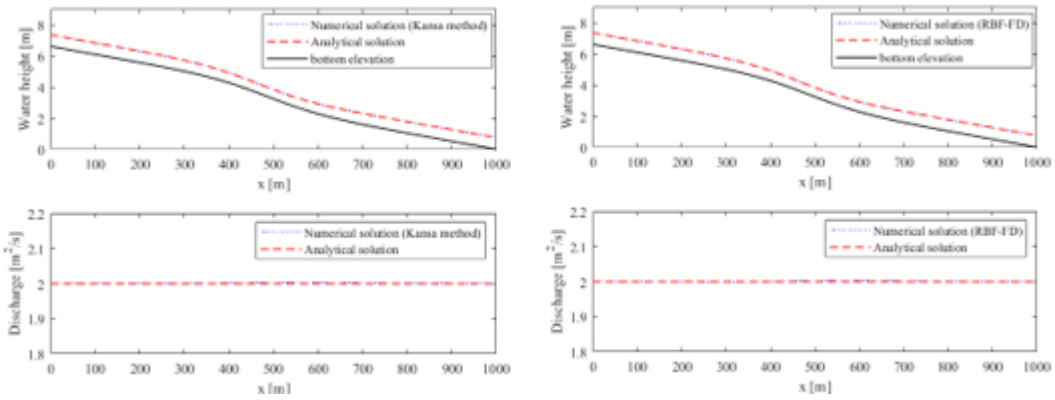
(c) RBF-PUM-QR method

Figure 10. L^2 error as a function of the viscosity coefficient for the ”Subcritical flow” problem for (a) Kansa, (b) RBF-FD, and (c) RBF-PUM-QR at T = 1500 seconds

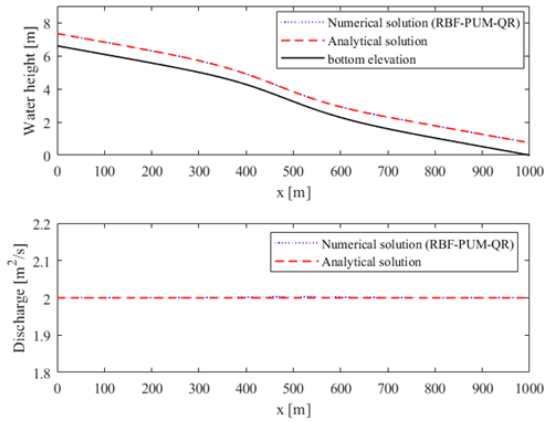


(a) (b)

Figure 11. (a) L^2 Errors of the RBF-FD method for solving ”Subcritical flow” problem with respect to the number of stencils using $\mu = 10^{-1}$. (b) L^2 error in terms of the shape parameter ε.

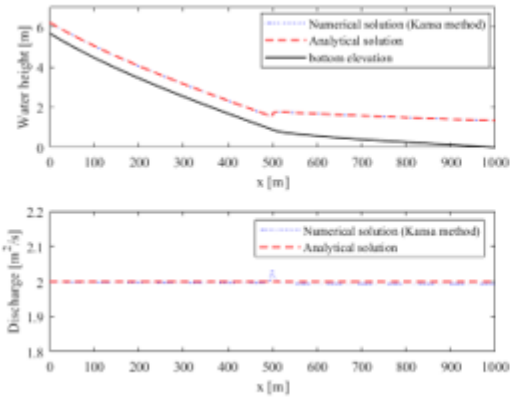


(a) Kansa method (b) RBF-FD method

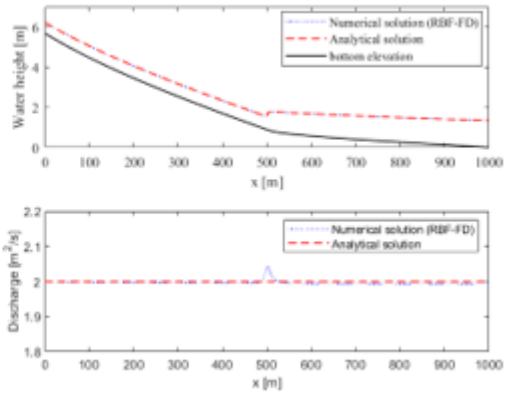


(c) RBF-PUM-QR method

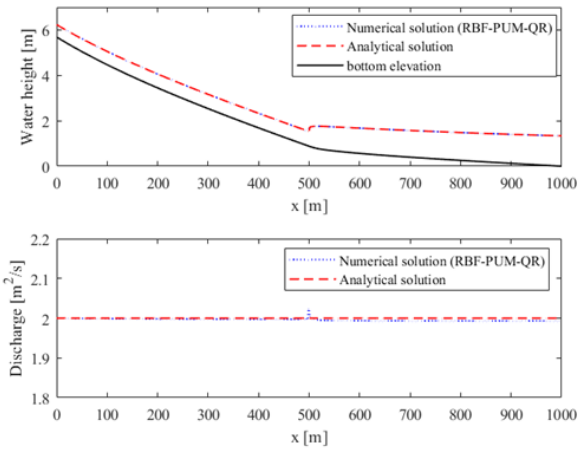
Figure 12. Supercritical flow using (a) Kansa, (b) RBF-FD, and (c) RBF-PUM-QR methods at T = 1500 seconds with Manning friction $n = 0.0218m^{-1/3}s$.



(a) Kansa method



(b) RBF-FD method



(c) RBF-PUM-QR method

Figure 13. Supercritical to subcritical flow using (a) Kansa, (b) RBF-FD, and (c) RBF-PUM-QR methods at T = 1500 seconds with Manning friction $n = 0.0218m^{-1/3}s$.

Refrences

[1] A.B. de Saint-Venant, Théorie du mouvement non permanent des eaux, avec application aux crues des rivières et a l'introduction de marées dans leurs lits, *Comptes rendus de l'Académie des Sciences de Paris 73*, 147--154 and 237-240 (1871).

[2] M.H. Chaudhry, *Open-channel flow*, Boston, MA: Springer US (2008).

[3] O. Delestre, C. Lucas, P.A. Ksinant, F. Darboux, C. Laguerre, et al., SWASHES: a compilation of Shallow Water Analytic Solutions for Hydraulic and Environmental Studies, *International Journal for Numerical Methods in Fluids 72(3)*, 269-300 (2013).

[4] Y. Xing, C.W. Shu, Solution of shallow-water equations using least-squares finite-element method, *Journal of Computational Physics 208(1)*, 206-227 (2005).

[5] S.J. Liang, J.H. Tang, M.S. Wu, High order finite difference WENO schemes with the exact conservation property for the shallow water equations, *Acta Mechanica Sinica 24*, 523-532 (2008).

[6] S.N. Kuiry, K. Pramanik, D. Sen, Finite Volume Model for Shallow Water Equations with Improved Treatment of Source Terms, *Journal of Hydraulic Engineering 134(2)*, 231-242 (2008).

[7] G.H. Duenas, A. Beljadid, A central-upwind scheme with artificial viscosity for shallow-water flows in channels, *Advances in Water Resources 96*, 323–338 (2016).

[8] Y.C. Hon, K.F. Cheung, X.Z. Mao, E.J. Kansa, Multiquadric Solution for Shallow Water Equations, *Journal of Hydraulic Engineering 125*, 524-533 (1999).

[9] S.M. Wong, Y.C. Hon, M.A. Golberg, Compactly supported radial basis functions for shallow water equations. *Applied Mathematics and Computation 127*, 79-101 (2002).

[10] S.M. Wong, Y.C. Hon, M.A. Golberg, Compactly supported radial basis functions for shallow water equations. *Applied Mathematics and Computation 127*, 79-101 (2002).

[11] E.J. Kansa, Multiquadrics-{A} scattered data approximation scheme with applications to computational fluid-dynamics-{II} Solutions to parabolic, hyperbolic and elliptic partial differential equations, *Computers and Mathematics with Applications 19(8)*, 147-161 (1990).

[12] A. Tolstykh, D. Shirobokov, On using radial basis functions in a finite difference mode with applications to elasticity problems, *Computational Mechanics 33*, 68-79 (2003).

[13] B. Fornberg, N. Flyer, *A primer on radial basis functions with applications to the geosciences*}, CBMS-NSF Regional Conference Series in Applied Mathematics, Society for Industrial and Applied Mathematics (SIAM), (2015).

[14] E. Ben-Ahmed, M. Sadik and M. Wakrim, Radial basis function partition of unity method for modelling water flow in porous media, *Computers and Mathematics with Applications 8(75)*, 2925-2941 (2018).

[15] V. Shcherbakov and E. Larsson, Radial basis function partition of unity methods for pricing vanilla basket options, *Computers and Mathematics with Applications 71(1)*, 185-200 (2016).

[16] E. Ben-Ahmed, M. Sadik and M. Wakrim, RBFPUM with QR Factorization for Solving Water Flow Problem in Multilayered Soil, *International Journal of Nonlinear Sciences and Numerical Simulation*, (2018).

[17] E. Ben-Ahmed, M. Sadik and M. Wakrim, A Stable Radial Basis Function Partition of Unity Method with d-Rectangular Patches for Modelling Water Flow in Porous Media, *Journal of Scientific Computing 84*, (2020).

[18] B. Fornberg, E. Lehto, Stabilization of rbf-generated finite difference methods for convective pdes, *Journal of Computational Physics 230*, 2270-2285 (2011).
[19] D. Stevens, H. Power, The Radial Basis Function Finite Collocation Approach for Capturing Sharp Fronts in Time Dependent Advection Problems, *Journal of Computational Physics 298*, 423-445 (2015).
[20] H. Ma, Chebyshev-Legendre super spectral viscosity method for nonlinear conservation laws, *SIAM Journal of Numerical Analysis 35*, 893-908 (1998).
[21] N. Flyer, E. Lehto, S. Blaise, G.B. Wright, A. St-Cyr, A guide to RBF-generated finite differences for nonlinear transport: Shallow water simulations on a sphere, *Journal of Computational Physics 231*, 4078-4095 (2012).
[22] N. Flyer, G.A. Barnett, L.J. Wicker, Enhancing finite differences with radial basis functions: Experiments on the Navier–Stokes equations, *Journal of Computational Physics 316*, 39-62 (2016).
[23] V. Shankar, G.B. Wright, A. Narayan, A Robust Hyperviscosity Formulation for Stable RBF-FD Discretizations of Advection-Diffusion-Reaction Equations on Manifolds, *SIAM Journal on Scientific Computing 42*, A2371-A2401 (2020).
[24] I. Tominec, M. Nazarov, Residual viscosity stabilized RBF-FD methods for solving nonlinear conservation laws, *Journal of Scientific Computing 94*, (2022).
[25] M. Al Nuwairan, E. Chaabelasri, Balanced Meshless Method for Numerical Simulation of Pollutant Transport by Shallow Water Flow over Irregular Bed: Application in the Strait of Gibraltar, *applied sciences 12*, 6849 (2022).
[26] Y. Liu, Y. Qiao, X. Feng, A stable radial basis function partition of unity method for solving convection-diffusion equations on surfaces, *Engineering Analysis with Boundary Elements 155*, 148-159 (2023).
[27] M. Dehghan, M. Abbaszadeh, The use of proper orthogonal decomposition (POD) meshless RBF-FD technique to simulate the shallow water equations, *Journal of Computational Physics 351*, 478-510 (2017).
[28] J. Borggaard, T. Iliescu, Z. Wang, Artificial viscosity proper orthogonal decomposition, *Mathematical and Computer Modelling 53*, 269-279 (2011).
[29] I. Babuska and J.M.~Melenk, The partition of unity method, *International Journal for Numerical Methods in Engineering 40(4)*, 727-758 (1998).
[30] V. Shcherbakov, E. Larsson, Radial basis function partition of unity methods for pricing vanilla basket options, *Computers and Mathematics with Applications 71*, 185-200 (2016).
[31] D. Shepard, A two-dimensional interpolation function for irregularly-spaced data, i*n: Proceedings of the 23rd ACM National Conference*, 517-524 (1968).
[32] H. Wendland, Piecewise polynomial, positive definite and compactly supported radial functions of minimal degree, *Advances in Computational Mathematics 4(1)*, 389-396 (1995).
[33] M.D. Buhmann, *Radial Basis Functions: Theory and Implementation*, Cambridge University Press, Cambridge (2003).
[34] H. Wendland, *Scattered Data Approximation*, Cambridge University Press, Cambridge (2005).

[35] C.A. Micchelli, Interpolation of scattered data: Distance matrices and conditionally positive definite functions, *Constructive Approximation 2*, 11-22 (1986).
[36] E. Larsson, V. Shcherbakov, A. Heryudono, A Least Squares Radial Basis Function Partition of Unity Method for Solving PDEs, *SIAM Journal on Scientific Computing 39*, A2538-A2563 (2017).
[37] A. Heryudono, E. Larsson, A. Ramage, L. von Sydow, Preconditioning for radial basis function partition of unity methods, *Journal of Scientific Computing 67*, 1089-1109 (2016).
[38] B. Forenberg, E. Larsson, N. Flyer, Stable computations with Gaussian radial basis functions, *SIAM Journal on Scientific Computing 33*, 869-892 (2011).
[39] B. Forenberg, C. Piret, A stable algorithm for flat radial basis functions on a sphere, *SIAM Journal on Scientific Computing 33*, 60-80 (2008).
[40] B. Fornberg, G. Wright, Stable computation of multiquadric interpolants for all values of the shape parameter. *Computers and Mathematics with Applications 48*, 853-867 (2004)
[41] Q. Liang, F. Marche, Numerical resolution of well-balanced shallow water equations with complex source terms, *Advances in Water Resources 32*, 873-884 (2009).
[42] T. Zhang, C.X. Zhan, B. Cai, C. Lin, X.M. Guo, An improved meshless artificial viscosity technology combined with local radial point interpolation method for 2D shallow water equations, *Engineering Analysis with Boundary Element 133*, 303-318 (2021).
[43] D. Satyaprasada, S.N. Kuiry, S. Sundar, A shock-capturing meshless method for solving the one-dimensional Saint-Venant equations on a highly variable topography, *Journal of Hydroinformatics 25*, 17-18 (2023).
[44] S. Sarra, A local radial basis function method for advection-diffusion-reaction equations on complexly shaped domains. *Applied Mathematics and Computation 218*, 9853-9865 (2012).
[45] A. Bermudez, M.E. Vazquez, Upwind methods for hyperbolic conservation laws with source terms, *Computers and Fluids 23(8)*, 1049-1071 (1994).
[46] N. Goutal, F. Maurel, Electriciteé de France Service Applications de l'Electriciteé et Environnement & Workshop on Dam-Break Wave Simulation, *In: Proceedings of the 2nd Workshop on Dam-Break Wave Simulation*, Direction des études et recherches, Electricité de France, (1997).
[47] E.F. Toro, *Shock-Capturing Methods for Free-Surface Shallow Flows*, Wiley and Sons (2001).
[48] Benkhaldoun. F, Seaid. M, A simple finite volume method for the shallow water equations, *Journal of Computational and Applied Mathematics 234*, 58-72 (2010).
[49] O. Delestre, *Simulation du ruisselement d'eau de pluie sur des surfaces agricoles*, PhD thesis (2010).
[50] I. MacDonald, *Analysis and computation of steady open channel flow*, PhD thesis, University of Reading - Department of Mathematics (1996).
[51] I. MacDonald, M.J. Baines, N.K. Nichols, P.G. Samuels. Analytic benchmark solutions for open-channel flows. *Journal of Hydraulic Engineering 123(11)*, 1041-1045 (1997).