

An adaptive finite point method for the shallow water equations

Enrique Ortega^{1,*}, Eugenio Oñate¹, Sergio Idelsohn^{1,‡} and Chinapat Buachart²

¹*International Center for Numerical Methods in Engineering (CIMNE), Universidad Politécnica de Cataluña, Edificio C1, Campus Norte, UPC, Gran Capitán, s/n, 08034 Barcelona, Spain*

²*School of Engineering and Technology, Asian Institute of Technology (AIT), P. O. Box 4 Klong Luang, Pathumthani 12120, Thailand*

SUMMARY

An adaptive Finite Point Method (FPM) for solving shallow water problems is presented. The numerical methodology we propose, which is based on weighted-least squares approximations on clouds of points, adopts an upwind-biased discretization for dealing with the convective terms in the governing equations. The viscous and source terms are discretized in a pointwise manner and the semi-discrete equations are integrated explicitly in time by means of a multi-stage scheme. Moreover, with the aim of exploiting meshless capabilities, an adaptive h -refinement technique is coupled to the described flow solver. The success of this approach in solving typical shallow water flows is illustrated by means of several numerical examples and special emphasis is placed on the adaptive technique performance. This has been assessed by carrying out a numerical simulation of the 26th December 2004 Indian Ocean tsunami with highly encouraging results. Overall, the adaptive FPM is presented as an accurate enough, cost-effective tool for solving practical shallow water problems. Copyright © 2011 John Wiley & Sons, Ltd.

Received 16 September 2009; Revised 1 March 2010; Accepted 20 January 2011

KEY WORDS: shallow water equations; adaptivity; meshfree methods; finite point method; tsunami simulation

1. INTRODUCTION

The shallow water equations (SWEs) model a wide range of geophysical flow problems involving the propagation of disturbances in incompressible fluids. Among their typical applications we can mention the analysis of wave propagation in oceans, lakes and coastal zones; the study of tidal motions, river and channel flows and the investigation of pollutant and sediment transport. The relevance of all these topics have in many fields of engineering and applied sciences, besides its social, humanitarian, economic and environmental implications, has motivated a growing interest from the part of researchers on the development of numerical techniques for solving the SWEs. Moreover, since the tragic episode of the 2004 Indian Ocean tsunami, significant efforts have been carried out to enhance forecasting models intended to develop early global alert systems and hazard mitigation policies; see for instance [1–3] and [4], where standards for assessing the numerical models employed in these systems are introduced. This new scenario is challenging and ever-demanding important advances especially in the fields of remote sensing, communications, real-time data analysis, modelling and simulation, hazard assessment and the design of global

*Correspondence to: Enrique Ortega, International Center for Numerical Methods in Engineering (CIMNE), Universidad Politécnica de Cataluña, Edificio C1, Campus Norte, UPC, Gran Capitán, s/n, 08034 Barcelona, Spain.

†E-mail: eortega@cimne.upc.edu

‡ICREA Research Professor at CIMNE.

action strategies. As regards modelling and simulation, faster and more reliable numerical models are needed and methods based on the SWEs have an important role to play.

The equations governing shallow water problems are obtained as a particular case of the principles of mass, momentum and energy conservation describing the behaviour of general fluid flows. Hence, the numerical techniques applied in shallow water problems closely resemble those techniques employed for solving Navier–Stokes equations. For instance, the hyperbolic nature of the inviscid SWEs makes it possible that a multitude of numerical schemes, originally designed in the context of gasdynamics, can be used for the treatment of convective terms (cf. [5]). In addition, the discretization of the viscous and source terms in SWEs can be performed by standard discretization techniques, as also occurs in Navier–Stokes equations. Concerning spatial discretization, conventional mesh-based methods such as Finite Differences (FD), Finite Volume (FV) and Finite Element (FE) have been successfully employed for solving the SWEs. Moreover, certain combinations between these methods have been also proposed in order to exploit the best characteristics of each particular technique, see for instance a combined FV-FE technique presented by Wang and Liu [6]. On the other hand, as happens in many fields of numerical computation, meshless methods arise as a feasible alternative to conventional discretization techniques.

The fact that meshless methods are conceptually simple and have potential advantages over conventional mesh-based techniques explains the scientific interest of many researchers in this area (cf. [7–9], for some reviews). Although practical meshless implementations have not been able so far of proving their efficiency and further efforts are needed in this sense, meshless methods have succeeded in demonstrating their capabilities in many fields of numerical computation. In the context of SWEs we can mention, for example, the Eulerian approach due to Ata and Soulaïmani [10], the Lagrangian implementation presented by Hon *et al.* [11] and the mixed FV-meshless application proposed by Cueto-Felgueroso *et al.* [12] with the purpose of achieving higher order discretizations on unstructured grids. Furthermore, there are particular applications of the SWEs which offer good opportunities for the development and promotion of meshless methods. Practical large-scale problems presenting complex topography and/or highly localized transient or stationary solution features are good examples of such applications. In general, these problems demand important discretization efforts in order to guarantee that acceptable levels of resolution and accuracy can be achieved with a reasonable computational cost. Several approaches have been proposed in the literature for dealing with this issues, see for instance the adapted unstructured-mesh generation technique presented by Scott *et al.* [13] and the nested-grid model due to Karim *et al.* [14]. In our view, meshless methods can make significant contributions to solve this kind of problems; thus, the present research is aimed at a preliminary exploration of some of the capabilities meshless techniques have in this field.

In the present work we deal with a meshless technique known as Finite Point Method (FPM) which was introduced by Oñate *et al.* [15–17]. In the FPM, the numerical approximation to the problem variables and their derivatives is based on a particular Weighted Least-Squares (WLSQ) procedure named Fixed Least Squares (FLS). The strong form of the governing equations is sampled at each point by replacing the continuous variables with their approximated counterparts and the resulting system of algebraic equations is obtained by means of a collocation technique. The FPM was presented in the literature towards the mid-nineties and since then it has been successfully applied to solve convective–diffusive problems, incompressible and compressible fluid flow problems [15–19] and solid mechanics problems [20] among others. More recently, some approaches towards improving the robustness of the FPM have been proposed, see for instance the work due to Boroomand *et al.* [21] and the technique we presented in [22], aimed at enhancing the WLSQ solution in three-dimensional clouds of points. As regards the exploitation of FPM capabilities, in the aforementioned work we have dealt with higher order Finite Point (FP) discretizations in a preliminary manner with the purpose of studying p -refinement possibilities. Moreover, concerning h -refinement, an application to solid mechanics problems has been presented by Perazzo *et al.* in [23] and we presented another adaptive methodology intended to deal with three-dimensional compressible aerodynamics problems in [24]. Following a similar line of research, in the present work we apply this adaptive technique to the solution of shallow water problems.

In the following, Section 2 outlines the basic aspects of the FP approximation. The shallow water equations and the finite point scheme we propose for its solution are presented in Sections 3 and 4 respectively, and the adaptive h -refinement technique is described in Section 5. Typical shallow water problems are solved in Section 6 with the aim of assessing the performance of the proposed methodology. Furthermore, an adaptive numerical simulation of the 2004 Indian Ocean tsunami is presented in Section 7 in order to show the possibilities the FPM has for solving real engineering problems as well as to exploit meshless features. Finally, the most relevant conclusions of this work are drawn in Section 8.

2. THE FPM

This section reviews the main aspects of the FPM with the objective of introducing the basic tools employed for discretizing the SWEs.

2.1. Numerical approximation on clouds of points

Suppose $u(\mathbf{x})$ is an unknown function defined in a closed domain $\Omega \in \mathfrak{R}^d$ ($d = 1, 2$ or 3), which is discretized by a set of points \mathbf{x}_i ($i = 1, n$) and Ω_i are local clouds of points covering Ω . Let each local cloud consist of np points resulting from a point \mathbf{x}_i called *star point* and a set of points \mathbf{x}_j surrounding it. Then, an approximation to the function $u(\mathbf{x})$ in Ω_i can be stated as

$$u(\mathbf{x}) \cong \hat{u}(\mathbf{x}) = \sum_{l=1}^m p_l(\mathbf{x}) \alpha_l = \mathbf{p}^T(\mathbf{x}) \boldsymbol{\alpha} \quad (1)$$

where $\mathbf{p}_j^T = [p^1(\mathbf{x}_j) \ p^2(\mathbf{x}_j) \ \dots \ p^m(\mathbf{x}_j)]$ is a complete polynomial base in \mathfrak{R}^d [22] and $\boldsymbol{\alpha}$ is an *a priori* unknown vector. The function $u(\mathbf{x})$ can be sampled at each point $\mathbf{x}_j \in \Omega_i$ according to Equation (1) and this procedure leads to the following system of equations:

$$\mathbf{u}^h = \begin{bmatrix} u_1^h \\ u_2^h \\ \vdots \\ u_{np}^h \end{bmatrix} \cong \begin{bmatrix} \hat{u}_1 \\ \hat{u}_2 \\ \vdots \\ \hat{u}_{np} \end{bmatrix} = \begin{bmatrix} \mathbf{p}_1^T \\ \mathbf{p}_2^T \\ \vdots \\ \mathbf{p}_{np}^T \end{bmatrix} \boldsymbol{\alpha} = \mathbf{P} \boldsymbol{\alpha} \quad (2)$$

where $u_j^h = u^h(\mathbf{x}_j)$ is the value of $u(\mathbf{x})$ at $\mathbf{x} = \mathbf{x}_j$ and $\hat{u}_j = \hat{u}(\mathbf{x}_j)$ is the approximated value at that point. Notice that system (2) becomes overdetermined when $np > m$ (usual choice for meshless approximations) and, consequently, an approximate solution must be sought. In the FPM, a particular WLSQ procedure known as FLS is adopted in such a way that its solution minimizes the following discrete functional:

$$\mathbf{J}(\mathbf{x}_i) = J_i = \sum_{j=1}^{np} \varphi_i(\mathbf{x}_j) [\hat{u}_j - u_j^h]^2 = \sum_{j=1}^{np} \varphi_i(\mathbf{x}_j) [\mathbf{p}_j^T \boldsymbol{\alpha} - u_j^h]^2 \quad (3)$$

in which $\varphi_i(\mathbf{x}_j) = \varphi(\mathbf{x}_j - \mathbf{x}_i)$ is a fixed compact-support weighting function centred on the star point of the cloud. This function is usually defined as

$$\varphi_i(x_j) = \frac{e^{-(d_j/\alpha)^k} - e^{-(\beta/\alpha)^k}}{1 - e^{-(\beta/\alpha)^k}} \quad (4)$$

where $d_j = \|\mathbf{x}_j - \mathbf{x}_i\|$, $\alpha = \beta/w$, $\beta = \gamma d_{\max}$ ($\gamma > 1$) and the parameters w , k and γ govern the functional shape of the weighting function; some guidelines for setting these free parameters are given in [22].

When the FLS procedure is applied, the approximation methodology is considerably simplified and its computational cost is reduced. However, FLS approximations lead to multivalued shape

functions depending on the cloud in which the approximation is computed, *i.e.* $\mathbf{N}_n(\mathbf{x}_j) \neq \mathbf{N}_m(\mathbf{x}_j)$ (subscripts m and n indicate neighbouring clouds of points). Therefore, the numerical approximation is globally and locally discontinuous and must be considered as valid only at the star point of the cloud where the weighting function is located. Hence, a collocation technique becomes the natural choice in the FPM.

The minimization of Equation (3) with respect to $\boldsymbol{\alpha}$ leads to the following system of equations:

$$(\mathbf{P}^T \boldsymbol{\phi}(\mathbf{x}) \mathbf{P}) \boldsymbol{\alpha} - (\mathbf{P}^T \boldsymbol{\phi}(\mathbf{x})) \mathbf{u}^h = \mathbf{0} \quad (5)$$

where $\boldsymbol{\phi}(\mathbf{x}) = \text{diag}(\varphi(\mathbf{x}_j - \mathbf{x}_i))$. Next, introducing the matrices

$$\mathbf{A} = (\mathbf{P}^T \boldsymbol{\phi}(\mathbf{x}) \mathbf{P}), \quad A_{kl} = \sum_{j=1}^{np} \varphi_i(\mathbf{x}_j) p_k(\mathbf{x}_j) p_l(\mathbf{x}_j) \quad (m \times m) \quad (6)$$

$$\mathbf{B} = (\mathbf{P}^T \boldsymbol{\phi}(\mathbf{x})), \quad B_{lj} = p_l(\mathbf{x}_j) \varphi_i(\mathbf{x}_j) \quad (m \times np)$$

it is possible to express system (5) as follows:

$$\mathbf{A} \boldsymbol{\alpha} = \mathbf{B} \mathbf{u}^h \quad (7)$$

As a fixed weighting function is chosen, the unknown coefficients α_j are constant in Ω_i . These coefficients can be found by solving

$$\boldsymbol{\alpha} = \mathbf{A}^{-1} \mathbf{B} \mathbf{u}^h \quad (8)$$

Then, replacing $\boldsymbol{\alpha}$ in Equation (1), the approximation to $u(\mathbf{x})$ at the star point of the cloud results in

$$\hat{u}(\mathbf{x}_i) = \underbrace{\mathbf{p}^T(\mathbf{x}_i) \mathbf{A}^{-1} \mathbf{B}}_{\mathbf{N}_i^T(\mathbf{x})} \mathbf{u}^h \quad (9)$$

where $\mathbf{N}_i^T(\mathbf{x}) = [N_1^i, N_2^i, \dots, N_{np}^i]$ is the shape function vector of point \mathbf{x}_i in Ω_i . The adoption of an FLS scheme, where matrices \mathbf{A} and \mathbf{B} are constant in Ω_i , simplifies the calculation of the shape function derivatives. Consequently,

$$\frac{\partial^l \mathbf{N}_i^T(\mathbf{x})}{\partial \mathbf{x}_k^l} = \frac{\partial^l \mathbf{p}^T(\mathbf{x}_i)}{\partial \mathbf{x}_k^l} \mathbf{A}^{-1} \mathbf{B} \quad (10)$$

and the approximation to the l -order unknown function derivatives at \mathbf{x}_i is given by

$$\frac{\partial^l \hat{u}(\mathbf{x}_i)}{\partial \mathbf{x}_k^l} = \frac{\partial^l \mathbf{N}_i^T(\mathbf{x})}{\partial \mathbf{x}_k^l} \mathbf{u}^h = \frac{\partial^l \mathbf{p}^T(\mathbf{x}_i)}{\partial \mathbf{x}_k^l} \mathbf{A}^{-1} \mathbf{B} \mathbf{u}^h \quad (11)$$

Notice that Equation (8) must be solved via matrix \mathbf{A} inversion because vector \mathbf{u}^h is not known in advance. Thus, depending on the spatial distribution of the local cloud of points, matrix \mathbf{A} can become highly ill-conditioned and be difficult to invert with accuracy. Numerical approximations in 2D clouds of points normally do not present this problem and, in general, can be solved satisfactorily by inverting matrix \mathbf{A} (this is the approach followed in this paper). However, 3D clouds often present numerical misbehaviours related to the spatial distribution of points, rendering it difficult to obtain a proper approximation. With the aim of alleviating these problems, the authors present an alternative approach for solving Equation (5) in [22]. This procedure, based on a QR factorization in conjunction with an iterative adjustment of the local approximation parameters, has demonstrated to increase notably the robustness of the FP approximation when complex 3D discretizations are undertaken.

2.2. Local clouds construction

The robustness of the local FP approximation can be enhanced working on the solution of the minimization problem; however, the characteristics of the geometrical support in which the latter is

based on still plays a major role. The procedure adopted in the present work for constructing local clouds of points can be summarized as follows. Given a point discretization of the computational domain and a set of normal vectors belonging to its boundary, a maximum (np_{\max}) and minimum (np_{\min}) allowable number of points in the cloud and an initial search radius are set. For each star point \mathbf{x}_i , all neighbour points within the search area are found and employed for constructing a local Delaunay grid. Only the first layer of nearest neighbours is retained and used to start the local cloud of points. Finally, admissible nearest points are added until the condition $np_{\min} \leq np \leq np_{\max}$ is fulfilled. It should be noticed that if a star point \mathbf{x}_i is located either over or close enough to a boundary, the admissible points included in its cloud must satisfy some additional conditions described in [24].

Including the first layer of Delaunay nearest neighbours of \mathbf{x}_i into the local cloud has proved to be particularly helpful for solving problems governed by conservation laws on distorted point discretizations. This procedure, which follows the lines proposed by Löhner *et al.* [19], enforces overlapping between neighbouring clouds of points and improves the quality of the local approximation. Moreover, the first layers of neighbour points provide useful data for implementing several computational procedures. In the present work such information is employed for the refinement technique presented in Section 5.

3. THE SWES

In shallow water flows the vertical length scale (depth) of the problem is much smaller than the wavelength of the disturbances travelling within the analysis domain and the flow movements are predominantly in a horizontal plane. Such problems can be studied as a particular case of the incompressible Navier–Stokes equations in which the acceleration in the depth direction is negligible and the pressure is supposed to be hydrostatic. Under these assumptions, the 3D Navier–Stokes equations can be integrated in depth and this procedure leads to a set of hyperbolic equations in a 2D horizontal plane where the flow variables are considered to be depth-averages. This model, which is extensively used in practice, constitutes a good approximation to a wide variety of geophysical flows and has the advantage of reducing modelling requirements considerably. A complete derivation of the 2D SWEs can be found in [25]. These can be written in a conservative differential form as

$$\frac{\partial \mathbf{U}}{\partial t} + \frac{\partial \mathbf{F}^k}{\partial x_k} + \frac{\partial \mathbf{G}^k}{\partial x_k} = \mathbf{Q} \quad (12)$$

where $k = 1, 2$ is the number of spatial dimensions of the problem, \mathbf{U} is the conservative variables vector, \mathbf{F}^k and \mathbf{G}^k are the advective and the diffusive fluxes vectors in the spatial direction x_k respectively and \mathbf{Q} is the source term. Vectors \mathbf{U} , \mathbf{F}^k and \mathbf{G}^k are given by

$$\mathbf{U} = \begin{bmatrix} H \\ Hu_1 \\ Hu_2 \end{bmatrix}, \quad \mathbf{F}^k = \begin{bmatrix} Hu_k \\ Hu_1 u_k + \delta_{1k} g(H^2 - h^2)/2 \\ Hu_2 u_k + \delta_{2k} g(H^2 - h^2)/2 \end{bmatrix}, \quad \mathbf{G}^k = \begin{bmatrix} 0 \\ -\bar{\tau}_{1k} \\ -\bar{\tau}_{2k} \end{bmatrix} \quad (13)$$

where $H = h + \eta$ is the total fluid depth defined as the sum of the resting depth h and the free-surface height η (measured from the resting fluid level). The k -component of the mean velocity vector \mathbf{u} is denoted as u_k , g is the gravity acceleration and δ_{ik} is the Kronecker delta. The depth-averaged viscous shear stresses $\bar{\tau}_{ik}$ in Equation (13) are given by

$$\bar{\tau}_{ik} = \nu H \left(\frac{\partial u_i}{\partial x_k} + \frac{\partial u_k}{\partial x_i} \right) \quad (14)$$

in which ν is the eddy kinematic viscosity of the fluid. Omitting effects due to free-surface stresses (e.g. wind) and variations in the atmospheric pressure, the source term \mathbf{Q} results [6]

$$\mathbf{Q} = \begin{bmatrix} 0 \\ g\eta S_{o1} - gHS_{f1} + HC_f u_2 \\ g\eta S_{o2} - gHS_{f2} - HC_f u_1 \end{bmatrix} \quad (15)$$

with C_f denoting the Coriolis parameter. The bottom friction terms S_{fk} can be defined [26] by

$$S_{fk} = \left(\frac{n}{k}\right)^2 \frac{u_k \|\mathbf{u}\|}{H^{4/3}} \quad (16)$$

where n is the Gauckler–Manning coefficient and k is a dimensional constant equal to 1.0 for SI units or 1.486 for US customary units. Finally, the bed slope terms in Equation (15) are computed as $S_{ok} = \partial h / \partial x_k$.

Notice that the solution of Equation (12) in a closed domain Ω with boundary Γ requires defining appropriate initial and boundary conditions. These will be discussed later in Sections 6 and 7.

4. THE FPM FLOW SOLVER

A wide collection of numerical schemes designed for dealing with hyperbolic conservation laws in the context of gasdynamics can be employed for solving the SWEs. In the present work, the general lines of the FP procedure we presented in [24] for the Euler equations are followed. This approach, which is based on the approximate Riemann solver of Roe [27] in conjunction with a limited MUSCL [28] extrapolation scheme, has been also successfully applied in the literature to SWEs; see for instance [6, 29, 30] and the references cited therein.

The semi-discrete counterpart of Equation (12) can be expressed as

$$\frac{\partial \hat{\mathbf{U}}_i}{\partial t} = -\frac{\partial \hat{\mathbf{F}}_i^k}{\partial x_k} - \frac{\partial \hat{\mathbf{G}}_i^k}{\partial x_k} + \hat{\mathbf{Q}}_i = \mathcal{F}_i + \mathcal{G}_i + \mathcal{Q}_i \quad (17)$$

where \mathcal{F}_i , \mathcal{G}_i and \mathcal{Q}_i are discrete approximations at point \mathbf{x}_i of the convective, diffusive and source term respectively. The calculation of these discrete terms is presented in the following section.

4.1. Discretization of the convective term

Taking into consideration the FP discretization described in Section 2 and after some manipulations [24], the discrete version of the convective term in Equation (17) can be written as

$$\mathcal{F}_i = -\frac{\partial \hat{\mathbf{F}}_i^k}{\partial x_k} = -2 \sum_{j \neq i} \frac{\partial N_j^i}{\partial x_k} [\tilde{\mathbf{F}}_{ij}^k - \tilde{\mathbf{F}}_i^k] \quad \forall j \in \Omega_i \quad (18)$$

where $\tilde{\mathbf{F}}_{ij}^k$ is the k -component of an unknown numerical flux vector computed at the midpoint of the line segment connecting the star point \mathbf{x}_i to another point $\mathbf{x}_j \in \Omega_i$ and $\tilde{\mathbf{F}}^k = \mathbf{F}^k(\mathbf{U}^h)$. Many possibilities exist in the literature for computing the numerical flux; in this work the approximate Riemann solver of Roe is employed. According to the latter, the Cartesian components of the numerical flux are obtained by

$$\tilde{\mathbf{F}}_{ij}^k = 1/2(\tilde{\mathbf{F}}_j^k + \tilde{\mathbf{F}}_i^k) - 1/2|\mathbf{A}_{\hat{n}}(\mathbf{U}_i^h, \mathbf{U}_j^h)|(\mathbf{U}_j^h - \mathbf{U}_i^h)\hat{n}^k \quad (19)$$

where \hat{n} is a versor in the direction of the vector $\mathbf{l}_{ji} = \mathbf{x}_j - \mathbf{x}_i$, $|\mathbf{A}_{\hat{n}}(\mathbf{U}_i^h, \mathbf{U}_j^h)|$ denotes the absolute value of the Roe matrix calculated in the same direction (cf. [6] for details) and the vector $(\mathbf{U}_j^h - \mathbf{U}_i^h)$

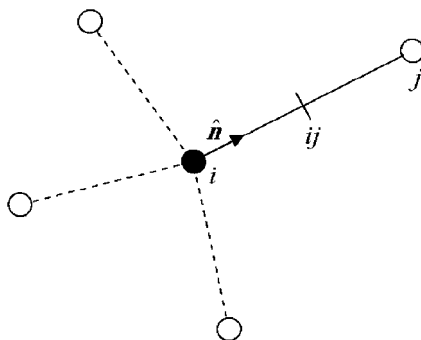


Figure 1. Stencil of points employed for discretizing the convective terms in a cloud of points.

is referenced to the resting depth h . The stencil of points employed in the derivation of expression (18) is shown in Figure 1.

As it is known, zero-order extrapolation for the variables at midpoint x_{ij} leads to a non-oscillatory first-order scheme. Aimed at increasing the spatial accuracy of the basis Roe solver, a MUSCL extrapolation procedure is applied following the general lines proposed in [19]. Thus, the zero-order extrapolations are replaced in Equation (19) by leftward and rightward higher order extrapolations $(\mathbf{U}_i^+, \mathbf{U}_j^-)$, enhancing the approximation to the numerical flux $\tilde{\mathbf{F}}_{ij}$. In addition, non-linear limiters are introduced into the extrapolation procedure in order to avoid numerical oscillations around discontinuities. In the present work, the numerical test cases presented in Sections 6 and 7 are solved using the Van Albada limiter [19].

It is important to point out that the conservative variables are referenced to the resting fluid depth when higher order extrapolations are computed. This avoids unwanted effects due to bathymetric data.

4.2. Discretization of diffusive and source terms

The finite point discretization of the diffusive flux vectors in Equation (17) is performed in a pointwise manner according to

$$\mathcal{G}_i = -\frac{\partial \hat{\mathbf{G}}_i^k}{\partial x_k} = -\sum_{j \in \Omega_i} \frac{\partial N_j^i}{\partial x_k} \tilde{\mathbf{G}}_j^k \quad (20)$$

where $\tilde{\mathbf{G}}^k = \mathbf{G}^k(\mathbf{U}^h)$. In a similar fashion, the discrete source term is computed by

$$\mathcal{Q}_i = \hat{\mathbf{Q}}_i = \sum_{j \in \Omega_i} N_j^i \tilde{\mathbf{Q}}_j \quad (21)$$

being $\tilde{\mathbf{Q}}^k = \mathbf{Q}^k(\mathbf{U}^h)$. In order to simplify the computation of the velocity gradient in the depth-averaged viscous stresses and the h gradient in the bed slope terms, the approximate value of a derivative at a given point x_i can be supposed to be equal to the value at that point ($\hat{u}_{,x_k}^i \approx (u_{,x_k}^i)^h$) without causing any negative impact on the accuracy of the resulting numerical scheme. In general, the fact that $(\cdot)^h$ and (\cdot) parameters become very closer in well-behaved FP approximations could explain this behaviour to a large extent because the shape functions tend to interpolate point data.

4.3. Discretization in time

The time derivative in the semi-discrete equations (17) is approximated in this work through a multi-stage Runge–Kutta scheme. Thus, the problem solution is explicitly advanced in time

from t^n to t^{n+1} by means of

$$\begin{aligned}\hat{\mathbf{U}}_i^{(0)} &= \hat{\mathbf{U}}_i^n \\ &\vdots \\ \hat{\mathbf{U}}_i^{(s)} &= \hat{\mathbf{U}}_i^n + \alpha_s \Delta t_i \mathbf{RHS}_i^{(s-1)} \\ &\vdots \\ \hat{\mathbf{U}}_i^{n+1} &= \hat{\mathbf{U}}_i^{(s_{\max})}\end{aligned}\quad (22)$$

being $\mathbf{RHS}_i^{(s-1)}$ the right-hand side of Equation (17) evaluated at an integration stage $s-1$, Δt_i the time step computed at point \mathbf{x}_i and α_s integration coefficients that depend on the number of stages employed (s_{\max}), see for instance [31]. The maximum allowable time step for each point x_i must be bounded due to stability requirements. This can be accomplished by setting

$$\Delta t_i = \mathcal{C} \min \left(\frac{\|\mathbf{I}_{ji}\|}{|\mathbf{u}_i \cdot \hat{\mathbf{n}}| + c_i}, \frac{\|\mathbf{I}_{ji}\|^2}{2\nu} \right) \quad \forall j \in \Omega_i \quad (23)$$

where \mathcal{C} denotes the Courant number, $c_i = \sqrt{gH_i}$ is the wave celerity and the rest of the variables have been defined before in Sections 3 and 4.1. Notice that local time steps Δt_i can be employed in order of increasing convergence speed when stationary solutions are sought whereas transient problems require advancing with a global time step (usually defined as the minimum of the local time steps Δt_i).

The difference between $(\cdot)^h$ and $(\hat{\cdot})$ parameters has already been pointed out in Section 2. Taking into account that $\mathbf{RHS}_i = f(\mathbf{U}_j^h) \nabla \mathbf{x}_j \in \Omega_i$, the following linear system has to be solved at the end of each integration stage:

$$\mathbf{M}\mathbf{U}^h = \hat{\mathbf{U}} \quad (24)$$

where $\mathbf{M} \in \mathfrak{N}^{n \times n}$ is the *mass matrix* of the system resulting from the assembly of the N_j^i coefficients. Although matrix \mathbf{M} can be considered to be diagonal in well-behaved approximations (see for instance the computation of viscous stresses in the previous section), no simplification is adopted as regards time integration to avoid undesirable cloud effects on this part of the numerical scheme which is essential. System (24) has excellent properties and can be solved by a few Gauss–Seidel iterations.

5. AN h -REFINEMENT TECHNIQUE FOR THE SWEs

Adaptive mesh refinement strategies play an important role in problems governed by SWEs. Real engineering problems in the field often involve facing moving waves with large wavelength and amplitude span, large-scale problems presenting highly localized topographical and/or solution characteristics and domains whose boundaries change in time among others. Adaptive mesh strategies are recommended in order to deal with these features, particularly in meshless contexts. In this work, the h -refinement FP technique developed in [24] for compressible aerodynamics problems is applied to the solution of shallow water problems. Next, the main aspects of the adaptive strategy are described.

5.1. Identifying the refinement needs

The first step towards achieving a discretization adapted to the problem solution consists in identifying those zones in the analysis domain where either refinement or unrefinement (coarsening)

is required. This is accomplished by means of the following *a posteriori* normalized indicator

$$\varphi_i = \frac{1}{\varphi_m} \sum_{j=1}^{n_{nei}} |I_{ji} \cdot (\nabla \eta_j - \nabla \eta_i)|, \quad i = 1, n \quad (25)$$

where n_{nei} is the number of points in the first layer of nearest neighbours of \mathbf{x}_i (already obtained in the local cloud construction stage), $I_{ji} = \mathbf{x}_j - \mathbf{x}_i$ is the vector linking each pair of points $(\mathbf{x}_i, \mathbf{x}_j)$, η is the fluid free-surface height and $\varphi_m = \max(\varphi_i)$. Then, the cloud corresponding to the star point \mathbf{x}_i is tagged for refinement when $\varphi_i > \varphi_{\max}$ and, conversely, point \mathbf{x}_i is marked to be removed from the computational domain if $\varphi_i < \varphi_{\min}$ (coarsening). Although parameters φ_{\max} and φ_{\min} must be set according to the problem under study, typical values for φ_{\max} can range from 0.1 to 0.5 and generally $10 > \varphi_{\max} / \varphi_{\min} > 20$ is chosen. Note that in particular cases, the proposed normalization in Equation (25) may cause some lack of sensitivity to relative small gradients in the flow field. The use of the local maximum for normalizing the indicator improves this behaviour.

5.2. Insertion and removal of points

When a cloud of points is selected to be refined ($\varphi_i > \varphi_{\max}$), the Voronoi vertices surrounding the star point \mathbf{x}_i are computed by means of its Delaunay grid of nearest neighbours. Next, a new point is set at each Voronoi vertex if it meets some geometrical conditions detailed in [24]. The coarsening procedure is carried out simply by deleting the point \mathbf{x}_i when $\varphi_i < \varphi_{\min}$ but the removal of points is restricted only to the existing points that have been inserted in prior refinement levels. This criterion guarantees a minimum appropriate geometrical support for the local approximation avoiding further verifications.

5.3. Update

Once the insertion and removal of points is finished, a few steps of a Laplacian smoothing are carried out on the affected area. This is particularly helpful when many points have been removed. After that, the clouds of points and the shape functions concerning the new points are constructed. In addition, the data concerning existing clouds of points affected by the insertion of new points or smoothing are reconstructed. Finally, the flow variables at the new points are obtained as an average of the variables at their previously existing nearest neighbours. This straightforward procedure can be enhanced if the local grid of Delaunay nearest neighbours is employed for interpolating variables between the newly adapted and the previous discretization.

6. NUMERICAL EXAMPLES

Several test cases are presented with the aim of illustrating the performance of the proposed methodology. The first two examples, which concern a dam-break problem and a forced tidal flow problem, are typical non-stationary test cases aimed at verifying the accuracy and convergence of the numerical scheme. The third example involves a steady flow past a channel with a backward step and is intended to assess the viscous behaviour of the scheme. Then, the effects of bed slope terms are evaluated by simulating the flow through a variable cross-section channel. At the end of this section, an adaptive test concerning a purely convective flow in a narrowing channel is presented aiming at analyzing the convergence of the *h*-refinement algorithm. For all the test cases presented, a two-stage Runge–Kutta scheme is employed for integrating the equations in time and the Courant number is set to $\mathcal{C} = 0.75$. Moreover, a third-order MUSCL extrapolation is employed in conjunction with the Van Albada limiter in order to increase the spatial accuracy of the underlying low-order scheme.

6.1. Dam-break flood problem

In this example we consider a straight channel which is initially divided by a dam. The water depth at the left side of the dam (H_L) is higher than the depth at the right of the dam (H_R),

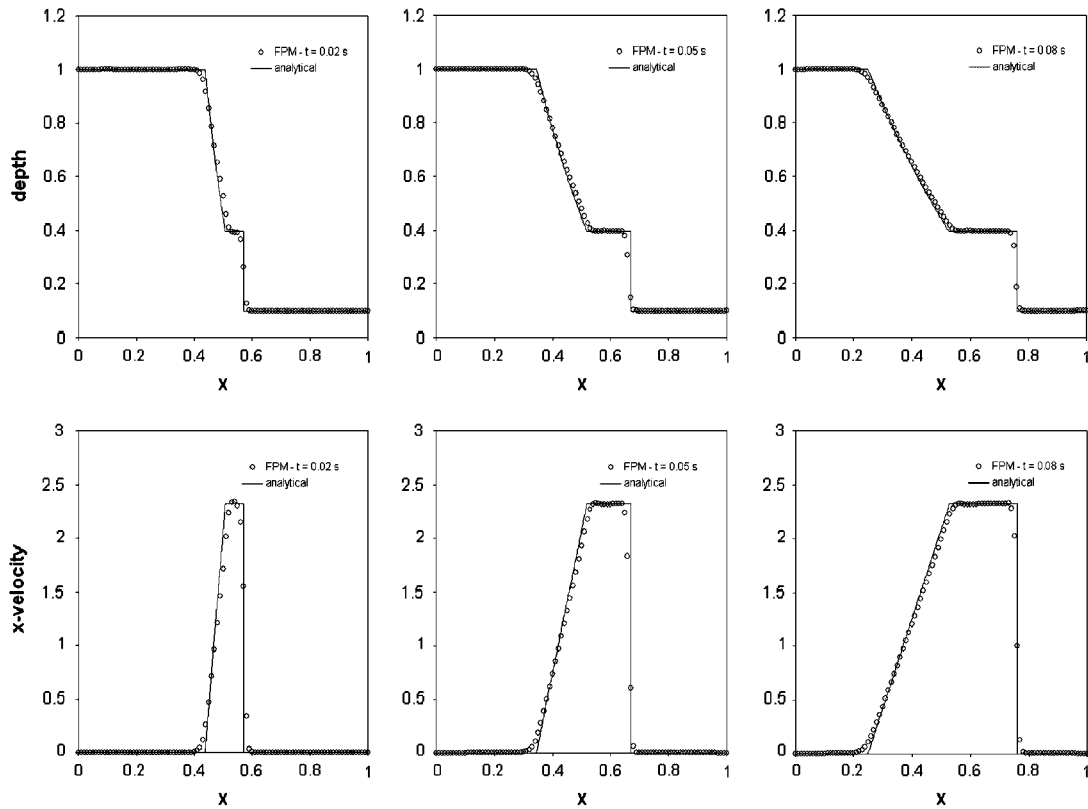


Figure 2. Dam-break problem $H_L/H_R = 10$. Comparisons between computed and analytical results.

the fluid is at rest at both sides and viscous effects are not considered. The simulation starts when the dam is instantly removed. This creates a bore wave moving from left to right and a depression wave propagating towards the left. This test, for which an exact solution of the SWEs is available, constitutes a challenging benchmark for verifying numerical solutions as well as assessing the temporal and spatial stability and shock-capturing capabilities of the numerical scheme. The computational domain is set to 1.0 m long (x -direction) and 0.5 m width (y -direction) and the dam is located at $x=0.5$ m. The domain is discretized by a structured distribution of 100×30 points in the x and y directions respectively ($n=3000$ points) and second-order spatial approximations are built in clouds where $12 \leq np \leq 17$. The initial conditions are $H_L = 1$ m, $H_R = 0.1$ m and $\mathbf{u}_L = \mathbf{u}_R = 0$. Water depth and velocity variations along the channel are computed at times $t=0, 0.02, 0.05$ and 0.08 s from the breaking of the dam. These results are compared with the analytical solutions due to Wu *et al.* [32] in Figure 2 and a close agreement is observed. Some snapshots of the fluid surface obtained at the same simulation times are displayed in Figure 3 (the coloured points are those employed for discretizing the problem).

6.2. Tide-driven flow in a long straight channel

The development of tidal waves along a channel is simulated and the numerical results are compared with the analytical linearized solution obtained by Lynch and Gray [33]. The problem is solved in a constant rectangular section channel having 8.8×10^5 m long and a resting water depth $h = 10$ m. The fluid is assumed to be inviscid and the channel walls are modelled as solid boundaries by enforcing slip-velocity conditions, with exception of the upstream boundary where a tidal forcing function $\eta(t) = \eta_0 \sin(\omega t)$ is prescribed ($\eta_0 = 0.1$ m and $\omega = 2\pi/T$ with $T = 12.4$ h). Moreover, the source term (15) is defined as $\mathbf{Q} = -\tau H[0, u_1, u_2]^T$ with the aim of introducing linear bottom dissipation and a friction coefficient $\tau = 5.0 \times 10^{-5}$ 1/s is specified. Intended to perform a spatial

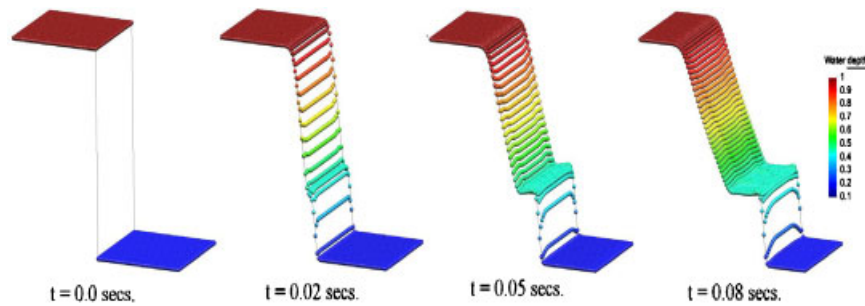


Figure 3. Water surface profiles computed at several times from the breaking of the dam ($H_L/H_R = 10$). Coloured points correspond to the discretization employed for solving the problem.

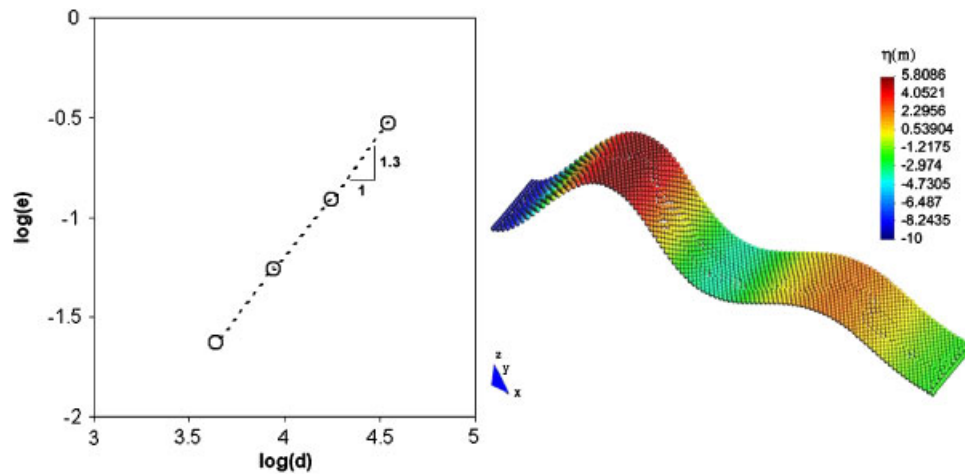


Figure 4. Tidal-driven flow problem. Left: spatial error norms of free-surface height. Right: surface elevation computed for a phase angle of 270° ($n=2161$ points). The vertical scale is magnified in order to highlight free-surface displacements.

(grid) convergence analysis four homogeneous distributions of 153, 554, 2161 and 8396 points (having approximately 25, 50, 100 and 200 uniformly spaced points along the channel length) are employed and second-order spatial approximations are computed in clouds where $12 \leq np \leq 17$. The simulation is initialized with $H = 10$ m and $u_1 = u_2 = 0$ and the time-dependent free-surface height $\eta(t)$ is imposed on the upstream channel boundary. After a few oscillation cycles, the steady-state periodic solution of the problem is reached. The order of accuracy of the numerical scheme is estimated by the following L_2 error norm:

$$e = \left(\frac{\sum_{i=1,N} (\eta_i^n - \eta_i^e)^2}{\sum_{i=1,N} (\eta_i^e)^2} \right)^{1/2} \quad (26)$$

where N denotes the number of points used in the norm computation, η_i^n is the free-surface height computed for a phase angle of 270° at a point x_i along the centreline of the channel and η_i^e is the analytical solution evaluated at the same point. Figure 4 depicts the error norm as a function of the average distance between points (d). There, the approximated order of convergence is $p = 1.3$, though second-order accuracy would be expected. To a large extent, this result may be explained as the spatial convergence rates in the FPM are very sensitive to factors such as the number and geometrical distribution of points in the local clouds and, particularly, to the weighting functions settings. Note that a similar trend can be observed for 3D Poisson's problems in [22]. In

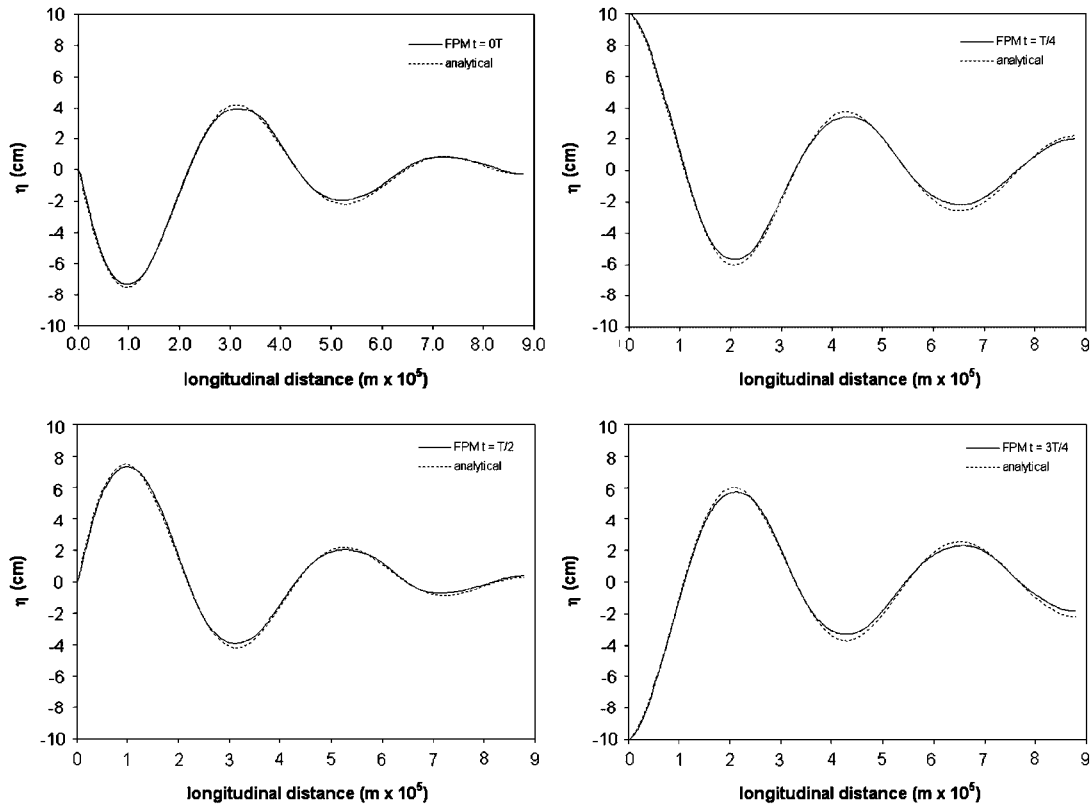


Figure 5. Development of tidal waves along a channel. Comparisons between numerical and analytical results at different times during one-period oscillation.

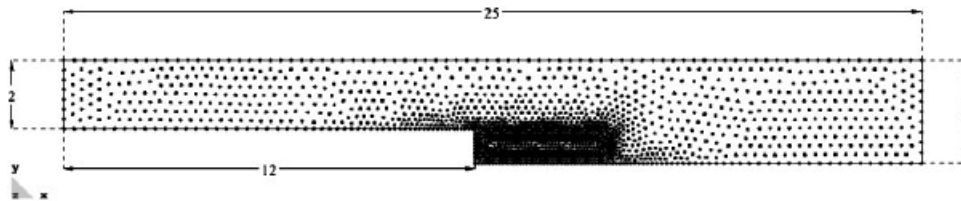


Figure 6. Top-view of the channel with a backward step displaying point discretization ($n = 1577$ points).

addition, small boundary effects on the error norms affecting the accuracy estimation should not be laid off.

Free-surface levels computed at different phase angles during one-period oscillation are compared with analytical results in Figure 5, where a good agreement can be observed for both, amplitude and location of the waves along the channel length. The numerical results are obtained for the third discretization employed in the convergence assessment presented before (approximately 100 points along the channel length).

6.3. Flow past a channel with a backward step

In this problem we compute a steady flow past a channel with a backward step with the aim of assessing the viscous behaviour of the numerical scheme. The step, located at a distance of 12 m from the upstream flow entry, has dimension 1 m and the downstream flow outlet is located 13 m ahead the step, as shown in Figure 6. The computational domain is discretized by a non-structured distribution of 1577 points and second-order spatial approximations are computed in clouds with

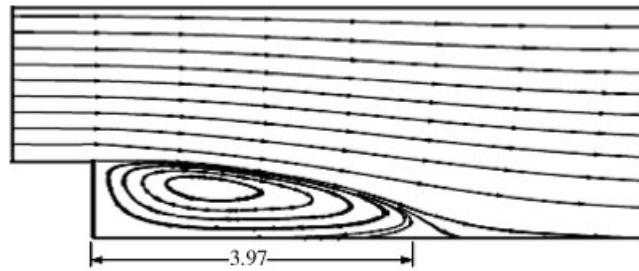


Figure 7. Top-view of the flow past a backward step showing computed reattachment length.

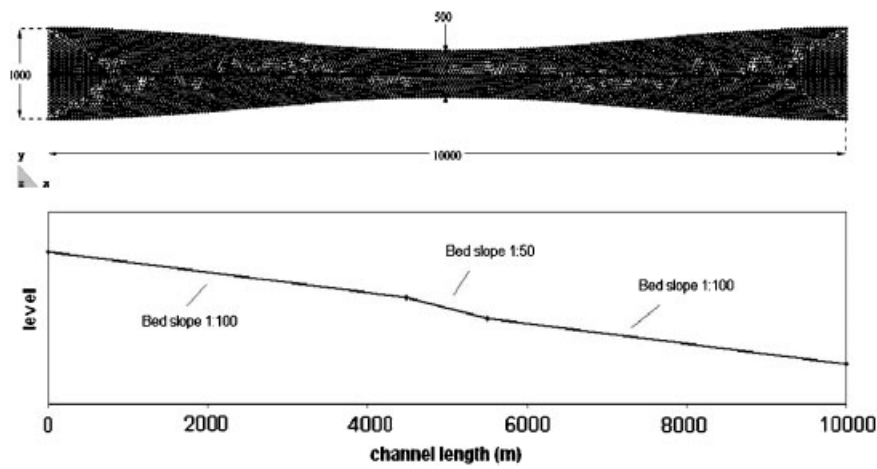


Figure 8. Top-view of the channel showing point discretization (5584 points) and bed slope variation.

$12 \leq np \leq 17$. The velocity at the inflow boundary and the fluid eddy viscosity are set in such a way that Reynolds number, based on the dimension of the step, is $Re = 73$ ($u = 0.5 \text{ m/s}$ and $\nu = 0.00685 \text{ m}^2/\text{s}$). A still water level of $H = 1 \text{ m}$ is prescribed at the outflow boundary and no-slip conditions are applied to the walls along the channel.

Next, the flow reattachment after the step is investigated and the results are compared with those presented by Wang and Liu [6]. Figure 7 shows that the computed reattachment length in this example is approximately 3.97 m and this result is very close to the 3.95 m reported by the aforementioned authors. According to the applied boundary conditions, a still water level of $H = 1 \text{ m}$ is obtained throughout the channel.

6.4. Flow along a channel with varying width and bed slope

This example, which involves an inviscid flow in a channel with a smooth constriction and variable bed slope, assesses the behaviour of bed slope terms in the numerical scheme. The channel is $1.0 \times 10^4 \text{ m}$ long and its width varies according to a cosine function, from 1000 m at the inflow boundary to 500 m at the throat, increasing afterwards to 1000 m at the outflow boundary. The bed slope is set to 1:100 (downward) along the channel excepting for the zone between $4500 \leq x \leq 5500 \text{ m}$ where a bed slope 1:50 is adopted (see Figure 8). The computational domain is discretized by a homogeneous distribution of 5584 points and second-order spatial approximations are computed in clouds where $12 \leq np \leq 17$. At the inflow boundary (left-side) a volumetric flow rate $Q = 2000 \text{ m}^3/\text{s}$ is prescribed whereas at the outflow boundary (right-side) the fluid depth is extrapolated from the interior flow field. In addition, slip-velocity condition is enforced on the walls along the channel.

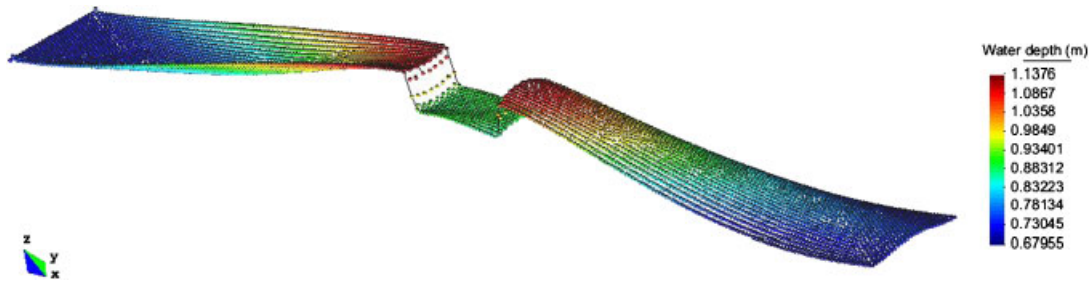


Figure 9. Computed FPM water surface profile along the channel (the vertical scale is augmented in order to highlight surface elevation).

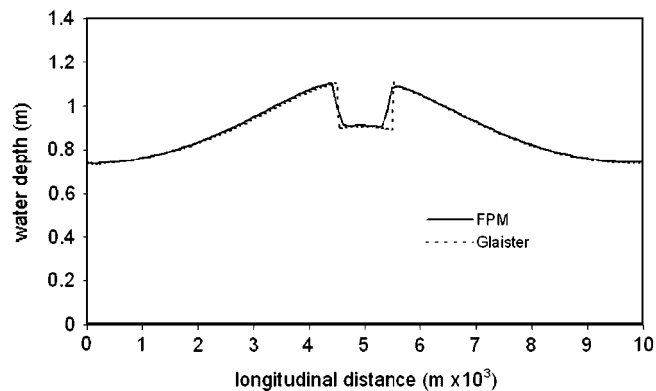


Figure 10. Water depth variation along the channel. Comparison between FPM and reference solution.

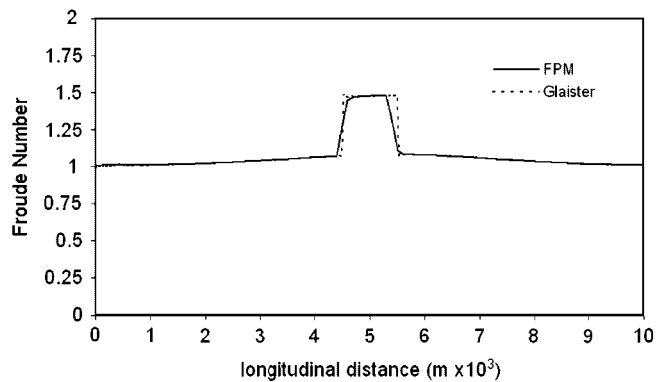


Figure 11. Froude number variation along the channel. Comparison between FPM and reference solution.

As open channel problems generally lack exact solutions, these problems have been extensively studied numerically and several reference solutions are available in the literature. In the present case, the FPM solution is compared with the numerical results by Glaister [26]. These are computed by using a one-dimensional FD flux-splitting scheme in 300 uniform cells. Figure 9 shows an FPM water surface profile where the hydraulic jumps induced by the changes in bed slope occurring at the central part of the channel can be observed.

Comparisons of water depth, Froude number and volumetric flow along the centre line of the channel are presented in Figures 10–12 respectively. A good agreement between the FPM results and the reference Glaister solution is obtained.

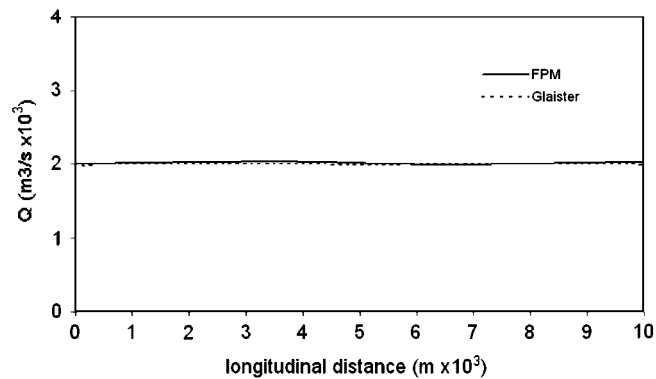


Figure 12. Volumetric flow along the channel. Comparison between FPM and reference solution.

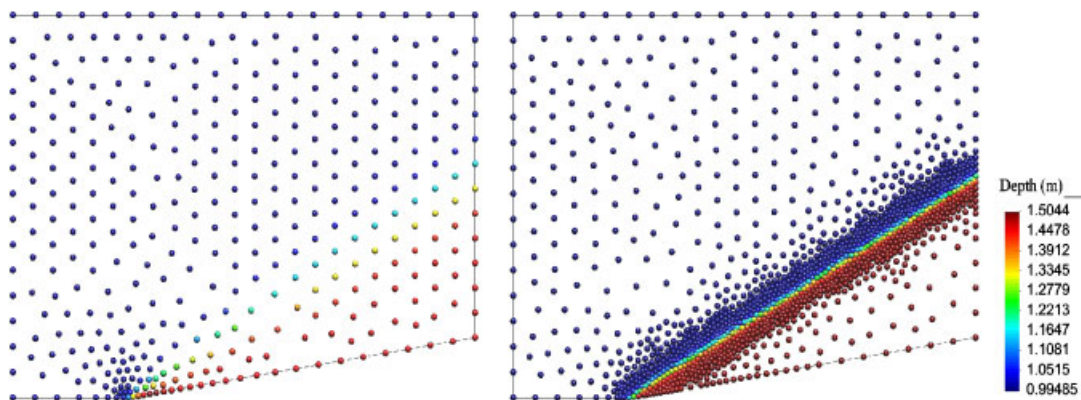


Figure 13. Top-view of the hydraulic jump problem displaying computed fluid depth. Left: initial coarse discretization ($n=395$ points). Right: final adapted discretization ($n=1404$ points).

6.5. An adaptive test case: the oblique hydraulic jump

The solution of a purely convective flow in a flat-bottom narrowing channel is intended to demonstrate the convergence of the h -refinement algorithm. The channel has 40 m long and an upstream entry 30 m width which is narrowed to the exit by a converging wall deflected through an angle $\theta=8.95^\circ$. The upstream boundary (left-side) conditions for this problem are $H_1=1$ m and $u_1=8.57$ m/s, which give a Froude number $Fr=2.736$. As the flow in this problem is supercritical no condition is applied to the downstream boundary (right-side) where the variables are let free. Friction along channel walls (top and bottom) is ignored and the fluid is considered to be inviscid. The computational domain is discretized by an initial coarse distribution of 395 points and second-order approximations are built in local clouds with $12 \leq np \leq 20$. The h -refinement strategy described in Section 5 is employed; the fluid depth H is used for evaluating the refinement indicator and the parameters φ_{\max} and φ_{\min} are set to 0.2 and 0.01 respectively. After 10 refinement levels, the problem discretization achieves 1404 points mainly located around the wave discontinuity. A top-view of the channel showing the initial and the final adapted point discretization is presented in Figure 13, where computed results for fluid depth across the channel are also shown. Next, the flow variables computed for the final adapted discretization at both sides of the jump discontinuity are presented in Figure 14. A close agreement with the exact solution of the problem presented by Alcrudo and Garcia-Navarro [34] ($H_2=1.5$ m, $|\mathbf{V}_2|=7.9556$ m/s and $\beta=30^\circ$) can be observed. Finally, the fluid depth obtained along a cut in the domain at different refinement levels is compared with the analytical solution in Figure 15. There, the convergence of successive adapted solutions to the analytical results becomes evident.

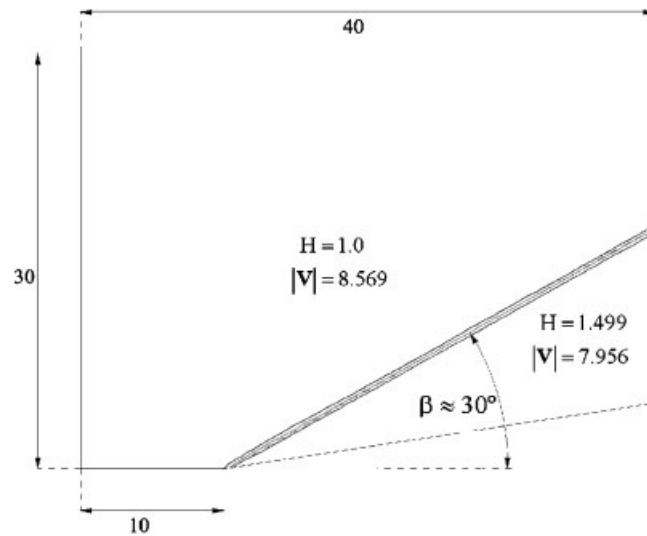


Figure 14. Depth contour plot computed for the final adapted discretization ($n = 1404$ points).

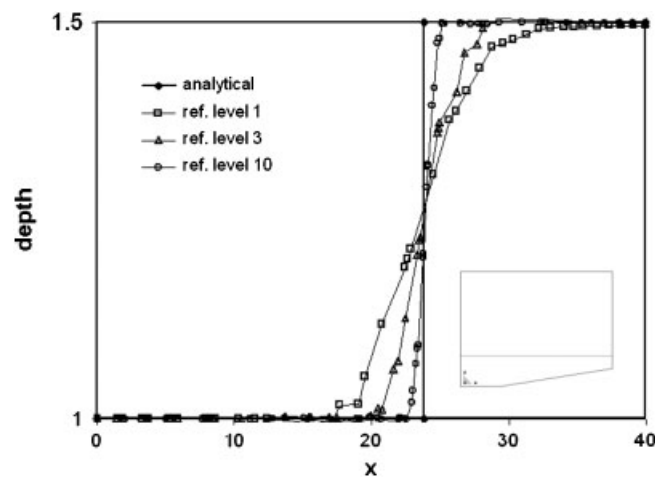


Figure 15. Convergence of successive adapted discretizations to the analytical solution of the hydraulic jump problem. The fluid depth is computed along a cut in the domain given by $0 \leq x \leq 40$ and $y = 8$.

7. ADAPTIVE SIMULATION OF THE 2004 INDIAN OCEAN TSUNAMI

The Indian Ocean earthquake, which struck the northern coast of Sumatra on 26th December 2004, generated an unprecedented tsunami with devastating consequences in terms of loss of human lives, infrastructure and environmental damage affecting mainly Indonesia, Sri Lanka, India and Thailand. Such a natural disaster made many research groups around the world turn their focus of attention towards developing improved early alert systems [1–3] in which propagation and inundation models play an essential role. These models, aimed at simulating the evolution of the tsunami after it is generated, must predict accurately the affected coastal locations, the height and velocity of the waves reaching the coastline, the arrival times and the resulting inundation areas (not always having enough input data available). Furthermore, computational time must be kept as low as possible to provide enough lead time for evaluating the hazards and executing proper mitigation actions. This is an important time constraint, which brings about the employment of

approximate analytical or semi-empirical propagation models; although most applications intended to carry out general accurate analyses heavily rely on solving SWEs (frequently interacting with real-time measurements and database information). In spite of the fact that important advances have been made in tsunami forecasting technology during the recent years, the current trends pursue for even more accurate models with improved capabilities and higher computational efficiency. Following these lines, we carry out a preliminary exploration of the possibilities the FPM has in this field by performing an adaptive simulation of the Indian Ocean tsunami, focusing on wave propagation towards the southern coast of Thailand.

In tsunami numerical simulation some essential aspects such as the initial and boundary conditions and the resolution of the topography and the bathymetric data must be carefully accounted for to achieve reliable results. First, initial conditions define the form of the excitation triggering the tsunami and determine the wave pattern which will be propagated throughout the analysis domain. Consequently, changes in the initial conditions influence the solution of the problem to a large extent. Unfortunately, the real conditions triggering a tsunami event are difficult to know with accuracy and that is why approximated models must be employed (especially within the context of early alert system). Second, boundary conditions must be in accordance with the mathematical behaviour of the equations. In addition, they should avoid unphysical reflections from open-sea boundaries and model the flow behaviour near the coastlines properly. Finally, the resolution of the topography and bathymetric data also plays an important role in the fidelity of the computational model and the numerical behaviour of the equations. In the following sections, the main implementation details in the present tsunami simulation are addressed.

7.1. Problem set-up

Aimed at modelling the propagation of the tsunami, the inviscid SWEs (12) are solved and Coriolis and bottom stress terms are taken into account. The initial domain discretization consists of a coarse non-structured distribution of 8240 points and second-order approximations are built in local clouds with $15 \leq np \leq 25$. A three-stage Runge–Kutta scheme is used to carry out time integration and a Courant number $\mathcal{C} = 0.75$ is adopted. In order to increase the scheme spatial accuracy, third-order MUSCL extrapolation is employed in conjunction with the Van Albada limiter. As regards the refinement strategy, the indicator is computed according to Equation (25) and the parameters φ_{\max} and φ_{\min} are set to 0.5 and 0.05 respectively. Starting from the initial excitation triggering the tsunami, successive refinement levels are performed at intervals of approximately 2.5 min (simulation time) and a maximum resolution of 2.5 km is set for the adapted discretizations. With the objective of simulating the propagation of the tsunami towards the southern coast of Thailand, particularly on Phuket Island, the total simulation time run in this example is approximately 150 min counting from the tsunami generation event. It is important to note that the successive retreat and rise cycles occurring after the tsunami hits the coastline are not simulated because this example employs an approximated topography, not reliable enough to reproduce this phenomenon with accuracy.

7.2. Topographical and bathymetric data

The analysis domain and the bathymetric data adopted in the present simulation are those employed by Kanok-Nukulchai and Nanakorn in [35], where the original bathymetric data obtained from the ETOPO2 database [36] are mapped into an FEM mesh for performing the computations. It should be noticed that the former study simplifies the computational domain by modelling all the small islands (e.g. Andaman Islands) including Phuket as a part of the ocean with a water depth of 5 m. In that case such procedure is justified by the fact that no local features around the coasts were sought and only the global wave propagation characteristics were of interest.

In the present simulation the bathymetric data from [35] are mapped into the finite point discretization obtained at each refinement level achieving both, bathymetry and topography of the computational model to be improved simultaneously. In order to map bathymetric data for each target point in the computational domain, the centroids of the elements in the background bathymetry mesh are used for searching the element where the target point lies on by using bins.

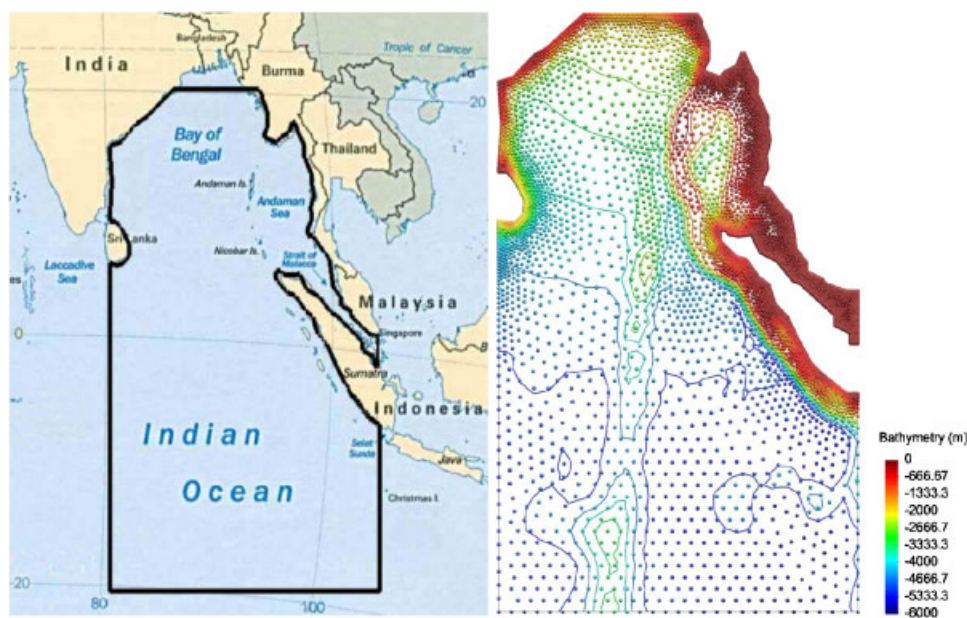


Figure 16. Tsunami simulation problem. Left: geographical delimitation of the analysis domain. Right: initial coarse point discretization displaying bathymetric data ($n = 8240$).

Then, the bathymetric data are interpolated to the target point using the standard shape functions of the underlying element. This procedure behaves efficiently even under large discretizations and a considerable speed-up is achieved by increasing the number of bins (notice that the bin data structure must be constructed only once, as the background bathymetry mesh is fixed). Next, Figure 16 shows the geographical delimitation of the computational domain and an example of bathymetric data computed for the initial coarse discretization ($n = 8240$).

7.3. Initial and boundary conditions

The 26th December 2004 tsunami was triggered by an undersea earthquake located 160 km west of the north coast of Sumatra, which reached a magnitude 9.0 on the Richter scale. The earthquake was generated by a subsidence fault of approximately 20 m width and 1000 km length in the confluence of the Eurasian and the Indo-Australian tectonic plates. After the incident, the location and extension of the fault as well as the vertical seabed displacements have been determined and several fault models have been proposed. The test case presented here adopts the model proposed by NORSAR [37], in accordance with that employed in [35]. Following this fault model, a sudden upward movement of the west side of the fault around 5 m and a downward movement of the east side around -3.5 m are considered. The water above the fault is supposed to move in a similar way (no-slip is assumed) and these displacements are applied as initial condition for the free-surface water height η . In addition, the initial velocity is considered to be zero everywhere in the analysis domain. Figure 17 depicts the initial free-surface height adopted in this example.

As regards boundary modelling, slip-velocity and non-reflective conditions are employed. The former are applied on all the coastlines in the computational domain by cancelling, at each point, the normal component of the flux vector. The latter, which are implemented through variables extrapolation from the inside of the domain, are intended to simulate open-sea boundaries avoiding undesirable wave reflections.

7.4. Numerical results

The numerical results computed at different simulation times from the initial excitation triggering the tsunami are shown in Figure 18. There, west and east travelling waves, in agreement with

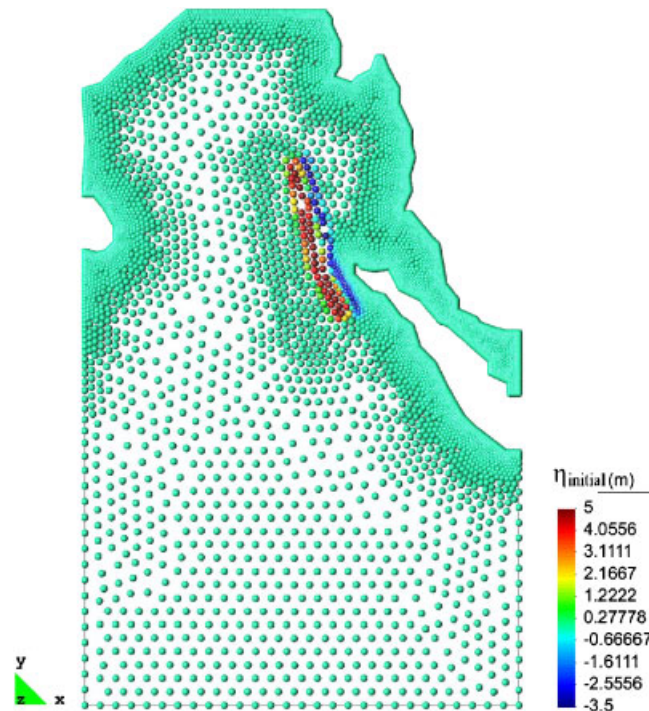


Figure 17. Tsunami simulation problem. Initial condition for the free-surface height η .

the leading north–south orientation of the subsidence fault, can be observed. The west moving waves travel faster than the east ones and exhibit longer wavelength and smaller amplitude as the ocean depth is higher on the west side of the fault. This fact explains the behaviour of the adapted finite point discretizations displayed in Figure 18, for which really few new points are added on the west side area when the simulation time goes beyond 30 min from the initial excitation. As regards east travelling waves, Figure 18 shows that the tsunami is very close to the north-west coast of Sumatra (Indonesia) around 15 min after the initial excitation. The timeline conected in [3] according to media reports indicates that the tsunami hit Banda Aceh (located in the northern coast of Sumatra) 17 min after the earthquake while several numerical results given in that reference also estimate around 20 min. The tsunami arrival time computed for Banda Aceh in the present example is approximately 19 min, which is consistent with the aforementioned results (notice that we consider the tsunami arrival time to occur for a zero free-surface height, just before the main wave rising up).

The time history of free-surface water level computed at Banda Aceh is shown in Figure 19, where a maximum wave height of around 11 m can be observed. This value is within the range of measurements collected by the United States Geological Survey (USGS) which reports wave heights between 5 and 13 m at Banda Aceh, increasing up to 30 m at locations along the west coast of the island [38].

Approximately 75 min after the earthquake, the east travelling waves approach the southern coast of Thailand. Focusing on the tsunami propagation towards Phuket Island, free-surface water levels are plotted in Figure 20 along a straight line going from the excitation area to Phuket, covering a distance of approximately 600 km.

As can be observed in Figure 20, the main tsunami wave decreases its wavelength and increases its amplitude as it approaches Phuket. For a simulation time of 90 min, the wave crest is located around 70 km from Phuket and the computed free-surface level is around 5 m. All these results follow the same trend presented in [35].

The evolution of water free-surface level computed in this example at Phuket location is depicted in Figure 21. The results show an ahead running depression wave causing the water level to recede

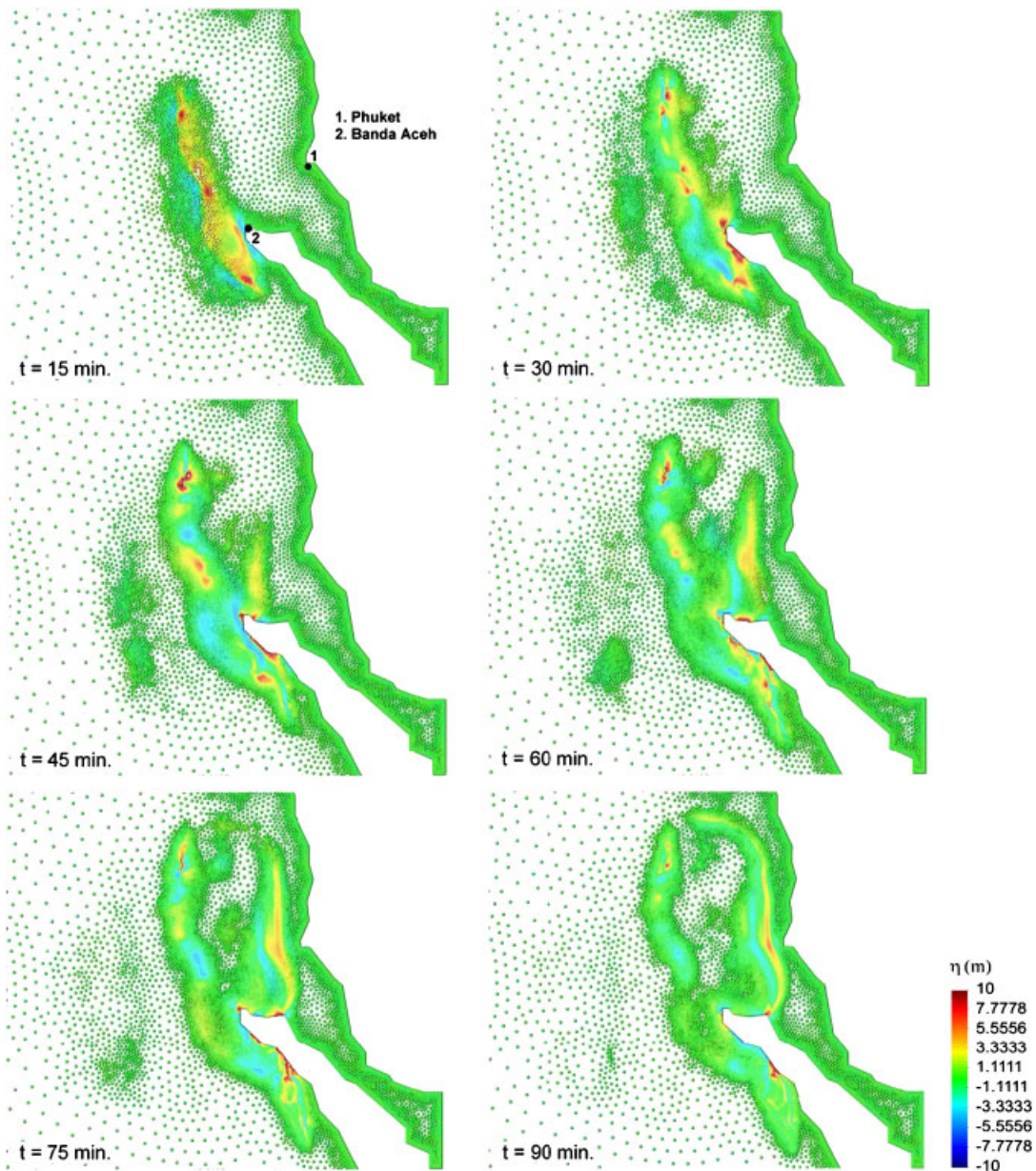


Figure 18. Adapted finite point discretizations computed at different simulation times from the Tsunami triggering event. The coloured points display the computed free-surface water height.

between 75 and 100 min simulation time. Around 105 min after the initial excitation, a sudden increase in the water level occurs due to the arrival of the positive tsunami wave and the water level rises for a period of 20 min, reaching its maximum height at around 125 min simulation time.

The tsunami arrival time computed in this example is compared in Table I with other results published in the literature. In spite of the fact that slight differences are observed, we found that, in general, the agreement is satisfactory and the differences in the reported values could be attributed to methodological factors such as the arrival criteria, the model resolution, the initial conditions and the sampled locations adopted in each particular work.

Figure 21 also shows a comparison between our results and a few point data extracted from elevation time series presented in [14] for the north-west coast of Phuket. Although the estimated time of events is pretty similar, the attained water levels exhibit some discrepancies. This fact

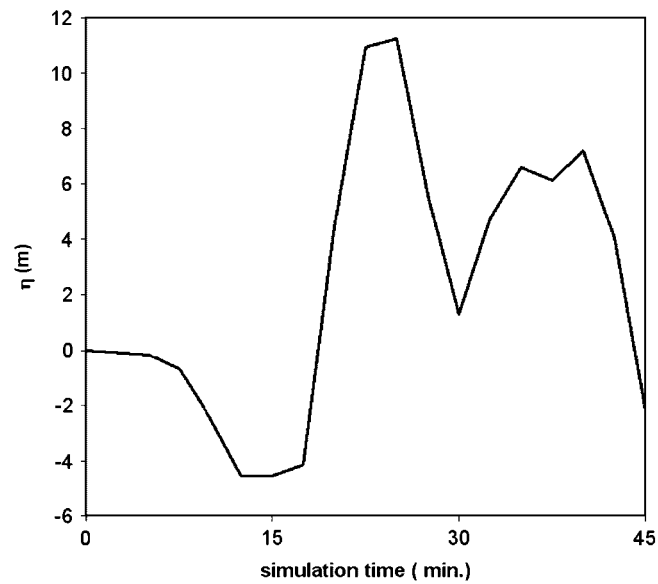


Figure 19. Approximated free-surface water level computed at Banda Aceh starting from the initial excitation.

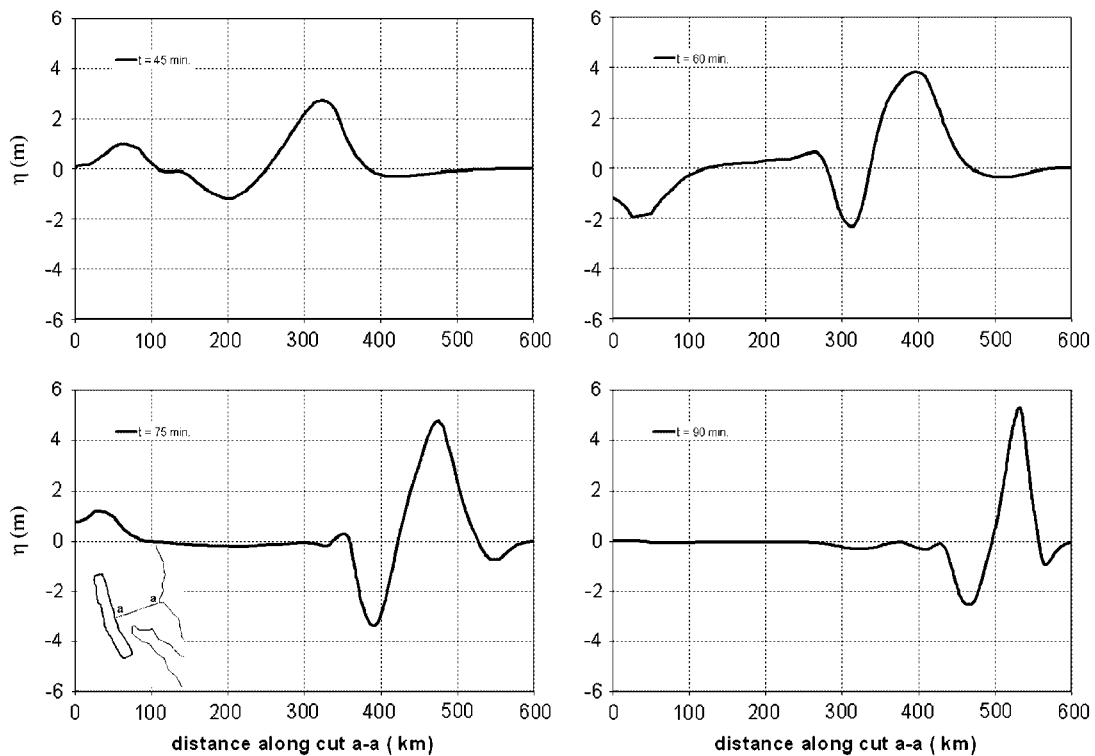


Figure 20. Free-surface water level along the path a-a computed at different times from the initial excitation.

is somehow predictable as the local topography of Phuket Island is not exactly resolved in our example. The present computation estimates the maximum wave height to be around 12 m, i.e. higher than that obtained in [14] but not incongruous with the range of values between 1.5 and 12 m the same authors found to occur in the island from south to north. A larger discrepancy exists in the negative wave, possibly due to the approximate topography we used or differences in

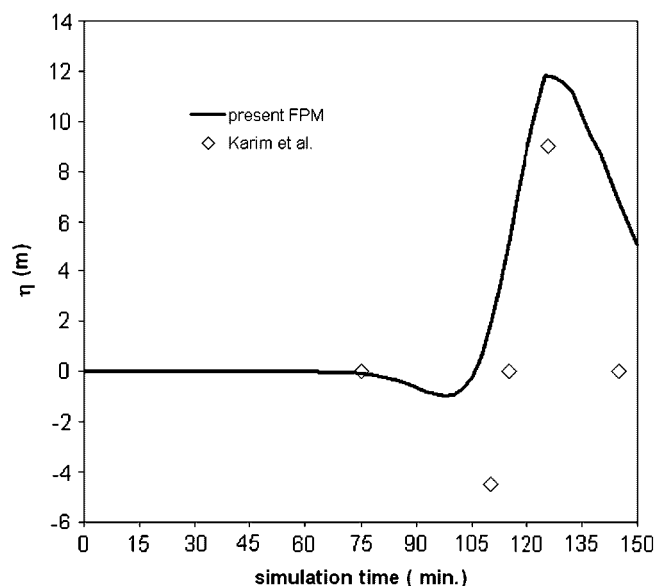


Figure 21. Computed free-surface water level at Phuket Island starting from the initial excitation.

Table I. Tsunami arrival times for Phuket Island.

		Arrival time (min)
Observed data	News reports [3]	105
	USGS [39]	90–120
Computed data	Present FPM	105
	Reported in [3]	90
	Karim <i>et al.</i> [14]	90–110

the initial conditions triggering the tsunami, mainly in the downward seabed displacements. We consider that, to a large extent, the latter could explain the disagreement in the attained negative water level observed in Figure 21.

Finally, concerning the performance of the present tsunami simulation, adapted discretizations are generated every several time steps with negligible time cost. The number of points in the domain does not surpass 50 000 at any time step; thus, the computational cost is kept low but not at the expense of the model resolution. This fact allows complete runs to be performed in a few minutes on computer desktops and renders the proposed adaptive finite point technique highly suitable for performing fast preliminary analyses.

7.5. Tsunami simulation. Concluding remarks

Tsunamis involve complex natural phenomena and consequently their numerical modelling is not at all a trivial task. In spite of the simplifications adopted in the present simulation, the numerical results are highly encouraging. However, some relevant issues such as an improved treatment of coastline boundary conditions, the employment of high-resolution models and the implementation of parallel computing strategies should be faced and dealt with in future investigations.

As regards the treatment of coastlines, we found that although the employment of slip conditions is a usual choice when dealing with inviscid models, it has not proved to be an accurate approximation to the real phenomenon and could cause an overestimation of the computed wave heights. Furthermore, numerical experiments performed with a viscous model revealed that if no-slip conditions on seashores are applied, free-surface heights result larger than those observed when slip boundaries are adopted in an inviscid model (although only a minor difference is

observed in the waves propagation features). A possible cause could be a strong restriction on the velocity vector but, anyway, both slip and no-slip boundaries are only a rough approximation to the flow behaviour at seashores. Consequently, the application of coastlines conditions in terms of the tangential viscous stress components seems to be a more appropriate choice when dealing with viscous models. In addition, other possibilities such as the modelling of dry areas by means of bottom stress terms could allow a better approximation for the real phenomenon occurring at the coastlines at the same time that this would be useful for computing inundation areas.

The other important aspects we consider should be faced in future developments are the employment of high-resolution models and the implementation of parallel computing strategies. On the one hand, both aspects result essential for satisfying the accuracy and computational efficiency requirements demanded in the field of tsunami forecasting. On the other hand, meshless capabilities would be profitable if adaptive high-resolution discretizations and parallel computing strategies (for which the FPM is quite well-suited) are taken into account. Finally, as long as parallel implementations are concerned, the employment of programming hardware such as GPUs offers promising applications, see for instance [39].

8. CONCLUSIONS

An FPM for solving shallow water equations has been presented. On the basis of a robust WLSQ approximation, an upwind semi-discrete scheme is constructed for each cloud of points. This methodology, in conjunction with an explicit multi-stage time integration scheme, has succeeded in solving accurately typical shallow water problems and several numerical examples presented in this work show evidence of this.

Practical shallow water problems usually involve large-scale domains having complex topography and/or exhibiting non-stationary solutions in which an elevated resolution in the discrete model is required. These features often present some difficulties to mesh-based approaches and can be easier handled if meshless techniques are applied. With the objective of exploiting meshless capabilities, we have presented an adaptive refinement technique which has been applied to the simulation of the tsunami event in the Indian Ocean on 26th December 2004. The numerical results obtained have been compared to those reported in the literature and the adaptive FPM has demonstrated to behave with considerable robustness, achieving reliable solutions with a low computational cost. All these facts are encouraging enough to do further research on potential possibilities that the FPM have for dealing with practical shallow water problems. In this respect, the efficient discretization of topographically complex models, the analysis of moving shorelines, the treatment of wet/dry zones within the computational domain in order to simulate inundation and the exploitation of parallel computing strategies are some of the challenging areas where the FPM could make really interesting and relevant contributions.

ACKNOWLEDGEMENTS

The authors express their gratitude to Prof. Worsak Kanok-Nukulchai from the Asian Institute of Technology and Dr Pruettha Nanakorn from the Sirindhorn International Institute of Technology for providing the data about the 26th December 2004 tsunami simulation employed in this work.

REFERENCES

1. Bernard EN, Mofjeld HO, Titov V, Synolakis CE, González FI. Tsunami: scientific frontiers, mitigation, forecasting and policy implications. *Philosophical Transactions of the Royal Society A* 2006; **364**:1989–2007.
2. Annunziato A. The tsunami assessment modelling system by the Joint Research Centre. *Science of Tsunami Hazards* 2007; **26**:70–92.
3. Annunziato A, Best C. The tsunami event. Analyses and models. *Institute for the Protection and Security of the Citizen. Joint Research Centre, European Commission*, 2005.

4. Synolakis CE, Bernard EN, Titov V, Kanoglu U, González FI. Standards, criteria and procedures for NOAA evaluation of tsunami numerical models. *NOAA Technical Memorandum OAR PMEL-135*, 2007.
5. Liska R, Wendroff B. Two-dimensional shallow water equations by composite schemes. *International Journal for Numerical Methods in Fluids* 1999; **30**:461–479.
6. Wang JW, Liu RX. Combined finite volume-finite element method for shallow water equations. *Computers and Fluids* 2005; **35**:1199–1222.
7. Belytschko T, Krongauz Y, Organ D, Fleming M, Krysl P. Meshless methods: an overview and recent developments. *Computer Methods in Applied Mechanics and Engineering* 1996; **139**:3–47.
8. Li S, Liu WK. Meshfree and particle methods and their applications. *Applied Mechanics Reviews* 2002; **55**:1–34.
9. Fries T, Matthies H. Classification and overview of meshfree methods. *Department of Mathematics and Computer Science, Technical University of Braunschweig. Inf. 2003-3*, 2004.
10. Ata R, Soulaïmani A. A stabilized SPH method for inviscid shallow water flows. *International Journal for Numerical Methods in Fluids* 2005; **47**:139–159.
11. Hon YC, Cheung KF, Mao XZ, Kansa EJ. A multiquadric solution for shallow water equation. *Journal of Hydraulic Engineering* 1999; **125**:524–533.
12. Cueto-Felgueroso L, Colominas I, Fe J, Navarrina F, Casteleiro M. High order finite volume schemes on unstructured grids using moving least squares reconstruction. Application to shallow water dynamics. *International Journal for Numerical Methods in Engineering* 2005; **65**(3):295–331.
13. Hagen SC, Horstmann O, Bennett RJ. An unstructured mesh generation algorithm for shallow water modeling. *International Journal of Computational Fluid Dynamics* 2002; **16**:83–91.
14. Karim MF, Roy GD, Ismail AIM, Meah MA. Numerical simulation of Indonesian tsunami 2004 along southern Thailand: a nested grid model. *International Journal of Mathematical, Physical and Engineering Sciences* 2009; **3**:1.
15. Oñate E, Idelsohn S, Zienkiewicz OC, Taylor RL, Sacco C. A stabilized Finite Point Method for analysis of fluid mechanics problems. *Computer Methods in Applied Mechanics and Engineering* 1996; **139**:315–346.
16. Oñate E, Idelsohn S, Zienkiewicz OC, Taylor RL, Sacco C. A finite point method for analysis of fluid mechanics problems. Applications to convective transport and fluid flow. *International Journal for Numerical Methods in Engineering* 1996; **39**:3839–3866.
17. Oñate E, Idelsohn S, Zienkiewicz OC, Fisher T. A finite point method for analysis of fluid flow problems. *Proceedings of the 9th International Conference on Finite Elements Methods in Fluids*, Venize, Italy, 15–21, 1995.
18. Oñate E, Sacco C, Idelsohn S. A finite point method for incompressible flow problems. *Computing and Visualization in Science* 2000; **3**:67–75.
19. Löhner R, Sacco C, Oñate E, Idelsohn S. A finite point method for compressible flow. *International Journal for Numerical Methods in Engineering* 2002; **53**:1765–1779.
20. Oñate E, Perazzo F, Miquel J. A finite point method for elasticity problems. *Computers and Structures* 2001; **79**:2151–2163.
21. Boroomand B, Tabatabaei AA, Oñate E. Simple modifications for stabilization of the finite point method. *International Journal for Numerical Methods in Engineering* 2005; **63**:351–379.
22. Ortega E, Oñate E, Idelsohn S. An improved finite point method for three-dimensional potential flows. *Computational Mechanics* 2007; **40**:949–963.
23. Perazzo F, Löhner R, Perez-Pozo L. Adaptive methodology for meshless finite point method. *Advances in Engineering Software* 2007; **39**:156–166.
24. Ortega E, Oñate E, Idelsohn S. A finite point method for adaptive three-dimensional compressible flow calculations. *International Journal for Numerical Methods in Fluids* 2009; **60**:937–971.
25. Zienkiewicz OC, Taylor RL, Nithiarasu P. *The Finite Element Method for Fluid Dynamics* (6th edn). Elsevier: Amsterdam, 2005.
26. Glaister P. Flux difference splitting for open-channel flows. *International Journal for Numerical Methods in Fluids* 1993; **16**:629–654.
27. Roe PL. Approximate Riemann solvers, parameter vectors and difference schemes. *Journal of Computational Physics* 1981; **43**:357–372.
28. Van Leer B. Towards the ultimate conservative difference scheme. V, A second order sequel to Godunov's method. *Journal of Computational Physics* 1979; **32**:101–136.
29. Ambrosi D. Approximation of shallow water equations by Roe's Riemann solver. *International Journal for Numerical Methods in Fluids* 1995; **20**(2):157–168.
30. Glaister P. Approximate Riemann solutions of the shallow water equations. *Journal of Hydraulic Research* 1988; **26**:293–306.
31. Jameson A. Artificial diffusion, upwind biasing, limiters and their effect on accuracy and multigrid convergence in transonic and hypersonic flows. *AIAA-93-3359*, 1993.
32. Wu C, Huang G, Zheng Y. Theoretical solution of dam-break shock wave. *Journal of Hydraulic Engineering* 1999; **125**:1210–1215.
33. Lynch DR, Gray WG. Analytic solutions for computer flow model testing. *ASCE Journal of the Hydraulics Division* 1978; **104**:1409–1428.
34. Alcrudo F, Garcia-Navarro P. A high-resolution Godunov-type scheme in finite volumes for the 2D shallow water equations. *International Journal for Numerical Methods in Fluids* 1993; **16**:489–505.

35. Kanok-Nukulchai W, Nanakorn P. Finite element modelling of tsunami propagation on the coast of Thailand. *Final Project Report Submitted to the Office of the Education Council, Thailand*, 2007.
36. National Geophysical Data Center (NGDC) web site. Available from: <http://www.ngdc.noaa.gov/mgg/bathymetry/relief.html> [16 June 2009].
37. NORSAR web site. Available from: <http://www.norsar.no/c-72-Tsunami-Studies.aspx> [16 June 2009].
38. U.S. Geological Survey (USGS) web site. Available from: <http://soundwaves.usgs.gov/2005/01/> [14 January 2009].
39. Liang WY, Hsieh TJ, Satria MT, Chang YL. A GPU-based simulation of tsunami propagation and inundation. *Lecture Notes in Computer Science*, vol. 5574. Springer: Berlin, Heidelberg, 2009.