



HAL
open science

Adaptive mesh refinement method. Part 2: Application to tsunamis propagation

Kévin Pons, Mehmet Ersoy, Frederic Golay, Richard Marcer

► **To cite this version:**

Kévin Pons, Mehmet Ersoy, Frederic Golay, Richard Marcer. Adaptive mesh refinement method. Part 2: Application to tsunamis propagation. 2016. hal-01330680v2

HAL Id: hal-01330680

<https://hal.science/hal-01330680v2>

Preprint submitted on 16 Jul 2016 (v2), last revised 3 Jul 2019 (v3)

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Adaptive mesh refinement method.

Part 2: Application to tsunamis propagation.

Kévin Pons, Mehmet Ersoy, Frédéric Golay and Richard Marcer

Abstract Numerical simulations of multi dimensional large scale fluid-flows such as tsunamis, are still nowadays a challenging and a difficult problem. To this purpose, a parallel finite volume scheme on adaptive unstructured meshes for multi dimensional Saint-Venant system is presented. The adaptive mesh refinement method is based on a block-based decomposition (called BB-AMR) which allows quick meshing and easy parallelization. The main difficulty addressed here concerns the selection of the mesh refinement threshold which is certainly the most important parameter in the AMR method. Usually, the threshold is calibrated according to the test problem to balance the accuracy of the solution and the computational cost. To avoid "hand calibration", we apply an automatic threshold method based on the decreasing rearrangement function of the mesh refinement criterion. This method is applied and validated successfully to the one and two dimensional non homogeneous Saint-Venant system through several tsunamis propagation test cases.

Kévin Pons

Principia S.A.S., Zone Athélia 1, 215 voie Ariane, 13705 La Ciotat cedex, France, e-mail: Kevin.Pons@principia.fr,
Université de Toulon, IMATH EA 2134, 83957 La Garde, France, e-mail: Kevin.Pons@univ-tln.fr

Mehmet Ersoy

Université de Toulon, IMATH EA 2134, 83957 La Garde, France, e-mail: Mehmet.Ersoy@univ-tln.fr

Frédéric Golay

Université de Toulon, IMATH EA 2134, 83957 La Garde, France, e-mail: Frederic.Golay@univ-tln.fr

Richard Marcer

Principia S.A.S., Zone Athélia 1, 215 voie Ariane, 13705 La Ciotat cedex, France, e-mail: Richard.Marcer@principia.fr

1 Introduction

Tsunamis are generally referred to any impulsed generated gravity wave. It can be generated by many sources. The most common tsunamis are a consequence of earthquake, landslide and volcanic explosion. Such events can displace a very large volume of water from its equilibrium. The displaced water mass, under the gravity action, attempts to come back to its equilibrium generating gravity waves. Depending of the tsunami source, the order of magnitude of a tsunami wave length is around ten kilometers. Since this horizontal length is much larger than the oceans depth, vertically integrated model are generally used for tsunamis propagation. One of the most simple model is the Saint-Venant model. These equations are a non-linear hyperbolic system of partial differential equations for which whose solutions may contain shock waves. For this model, the characteristic propagation speed in any direction, is

$$c = \sqrt{gh(x,t)}$$

where $h(x,t)$ stands for the water elevation above the bathymetry $Z(x)$ and $x \in \mathbb{R}^2$. These equations, being derived under the assumption of a hydrostatic pressure field (see for instance [11, 20, 7]), are well-known to be non-dispersive. It is also interesting to highlight that close to the shore, the shoaling phenomenon can be observed (i.e. whenever the wave starts to compress horizontally and grows vertically). For high shoaling effect, the non-linearities may become preponderant and the wave can break down. Because of the unique vertical evaluation, the solution of the Saint-Venant system cannot take into account such a solution. Nevertheless, a breaking wave can be represented by a shock solution. More precisely, the speed of propagation and the amplitude of the breaking wave are rather well-represented through the shock wave ones, see for instance [5]. Therefore, the propagation of a tsunami can be described quite accurately by the Saint-Venant equations.

However, solving accurately this model over very large ranges in spatial scale inexorably leads to heavy computational time [23, 24, 1]. Thus, Adaptive Mesh Refinement (AMR) methods allows, in principle, to solve in a reasonable CPU time these equations adjusting the computational effort locally to maintain a uniform level of accuracy. On one hand, the overall performance of the method can be improved in a parallel framework and more specifically with a Block-Based AMR method [14]. The BB-AMR technique provides an efficient control of the shared memory leading to well-balanced computational time between cores by domain-like decomposition. Nevertheless, data locality is critical to obtain good performance since the memory access times are not uniform and may become expensive. Consequently, the re-meshing step has to be carefully managed. Taking advantage of such a block-based structure (as in domain decomposition), we can define two different time-step: the first one is based on the CFL condition (i.e computed through the finest cells) while the second one is defined at the level of the block. The stability of the scheme is thus respected for any time and the re-meshing cost is minimized. On the other hand, the overall accuracy of the AMR method can be improved using an Automatic Mesh Refinement Threshold (AMRT) [22]. The AMRT technique

provides an efficient reading of the mesh refinement criterion yielding to an automatic threshold which avoids unnecessary mesh refinement. As a consequence, only pertinent regions are refined which may lead to a non negligible gain of computational time when compared to a fixed threshold. The AMRT method is based on the decreasing rearrangement of the mesh refinement criterion function and studied in details in [22]. It yields to a *parameterless* AMR method. Let us highlight that the most of the AMR methods are based on a "user calibrated" threshold α depending on the test case (see for instance [8]).

In this paper, we present a multi dimensional finite volume numerical scheme to solve accurately and fast non-linear hyperbolic systems of conservation laws in the context of *parameterless* BB-AMR framework. The AMR method has been first presented by Ersoy et al. [8] for the one-dimensional gas dynamics equations for ideal gas and extended to the BB-AMR method in the case of the two and three dimensional two-fluids flows [14, 1]. Finally, the AMRT has been first presented by Pons and Ersoy [22].

The first section of the paper is devoted to the presentation of the model governing equations (Sect. 2.1) and the finite volume approximation (Sect. 2.2). The second section is dedicated to the summarized presentation of the multi dimensional finite volume solver on unstructured meshes with an Automatic Mesh Refinement Threshold (Sect. 3.1) and a Block-Based Adaptive Mesh Refinement technique (Sect. 3.2). The model confrontation against experiments and the numerical validation is presented in the last section (Sect. 4). The first test case (Sect. 4.1) concerns the flume experiments conducted at Oregon State University [26, 25]. These experiments involve the propagation, run-up and reflection of high amplitude solitary waves on a reef. The second test case (Sect. 4.2) is the propagation of a solitary over a complex three dimensional shallow shelf [19]. In this experiment, the propagation, run-up, drying and flooding phenomena are involved. The last test case (Sect. 4.3) concerns the Monai-Walley tsunami run-up onto a complex three dimensional beach [17].

2 The Saint-Venant equations

As motivated in the introduction, the Saint-Venant system is one the simplest depth-averaged model able to describe accurately the propagation of a tsunami. In this section, we first present the Saint-Venant equations and we recall its properties. Then, we present the multi dimensional finite volume approximation. Unless otherwise indicated, bold characters are used for vector notation.

2.1 Description of the model

The two dimensional non linear Saint-Venant system is

$$\begin{aligned} \partial_t h + \operatorname{div}(h\mathbf{u}) &= 0 \\ \partial_t(h\mathbf{u}) + \operatorname{div}\left(h\mathbf{u}^2 + \frac{g}{2}h^2\mathbf{I}\right) &= -gh\nabla Z \end{aligned} \quad (1)$$

where the unknowns $h(x, t)$ and $\mathbf{u}(x, t) = (u_1, u_2)$ are respectively the height of the water and the depth-averaged velocity of the water at a space-time point (x, t) , $x = (x_1, x_2) \in \mathbb{R}^2$ is the space coordinate, $t > 0$ is the time, g is the gravitational constant $g \approx 9.81\text{m/s}^2$, $Z(x)$ is the bathymetry term and \mathbf{I} is the Identity matrix.

The three equations above express respectively, the conservation laws of mass and momentum in x_1 and x_2 direction driven by the fluxes

$$\mathbf{f}_1(\mathbf{w}) = \begin{pmatrix} hu_1 \\ hu_1^2 + \frac{g}{2}h^2 \\ hu_1u_2 \end{pmatrix} \text{ and } \mathbf{f}_2(\mathbf{w}) = \begin{pmatrix} hu_2 \\ hu_1u_2 \\ hu_2^2 + \frac{g}{2}h^2 \end{pmatrix}$$

and forced through the source

$$\mathbf{G}(x, \mathbf{w}) = \begin{pmatrix} 0 \\ -gh\partial_{x_1}Z \\ -gh\partial_{x_2}Z \end{pmatrix}$$

where the conservative variable \mathbf{w} is

$$\mathbf{w}(x, t) = \begin{pmatrix} h \\ hu_1 \\ hu_2 \end{pmatrix}.$$

In its vectorial form, System (1) reads

$$\partial_t \mathbf{w} + \operatorname{div} \mathbf{f}(\mathbf{w}) = \mathbf{G}(x, \mathbf{w}) \quad (2)$$

where $\mathbf{f}(\mathbf{w}) = (\mathbf{f}_1(\mathbf{w}), \mathbf{f}_2(\mathbf{w}))$.

System (2) can be also written in a quasi-conservation form as follows

$$\partial_t \mathbf{w} + \sum_{i=1}^{i=2} A_i \partial_{x_i} \mathbf{w} = \mathbf{G}(x, \mathbf{w}) \quad (3)$$

where A_i is the Jacobian matrix in the x_i direction

$$A_i = \frac{\partial \mathbf{f}_i}{\partial \mathbf{w}}.$$

From a numerical viewpoint, the two dimensional non conservative form (3) are less useful than in the one dimensional case since a simultaneous diagonalisation of A_1 and A_2 is not possible. Therefore the most of the schemes are based on a normal flux formulation as performed in this paper (see Sect. 2.2). In particular, in the case of Cartesian meshes, the flux formulation reduces to the one dimensional flux in the $(1, 0)$ and $(0, 1)$ directions. Therefore, without loss of generality, for any

given control volume $\mathbf{V} \subset \mathbb{R}^2 \times [0, T]$ of outward unit normal vector $\mathbf{n} = (n_1, n_2)$, the Saint-Venant system (2) satisfies the following properties

Theorem 1.

1. System (2) is strictly hyperbolic on the set $\{h(x, t) > 0\}$ where the eigenvalues are

$$\lambda_1(h, \mathbf{u}) = \mathbf{u} \cdot \mathbf{n} - \sqrt{gh}, \quad \lambda_2(h, \mathbf{u}) = \mathbf{u} \cdot \mathbf{n}, \quad \lambda_3(h, \mathbf{u}) = \mathbf{u} \cdot \mathbf{n} + \sqrt{gh},$$

and the right eigenvectors are

$$\begin{aligned} \mathbf{r}_1(h, \mathbf{u}) &= \left(1, u_1 - \sqrt{gh}n_1, u_2 - \sqrt{gh}n_2\right)^T, \\ \mathbf{r}_2(h, \mathbf{u}) &= \left(0, -\sqrt{gh}n_2, \sqrt{gh}n_1\right)^T, \\ \mathbf{r}_3(h, \mathbf{u}) &= \left(1, u_1 + \sqrt{gh}n_1, u_2 + \sqrt{gh}n_2\right)^T. \end{aligned}$$

2. For smooth solutions, the mean velocity u satisfies:

$$\partial_t \mathbf{u} + \mathbf{u} \cdot \nabla \mathbf{u} + g \nabla (h + Z) = 0. \quad (4)$$

3. For smooth solutions, the still water steady state, i.e. for $\mathbf{u} = 0$, reads

$$gh + gZ = \text{constant}. \quad (5)$$

4. System (2) admits a mathematical entropy :

$$E(h, \mathbf{u}, x) = \frac{h|\mathbf{u}|^2}{2} + \frac{gh^2}{2} + ghZ(x)$$

which satisfies the entropy (energy) relation

$$\partial_t E(h, \mathbf{u}, x) + \text{div} \left(\left(E(h, \mathbf{u}, x) + \frac{gh^2}{2} \right) \mathbf{u} \right) \leq 0 \quad (6)$$

where $|\mathbf{u}| := u_1^2 + u_2^2$.

Proof. Let $\mathbf{V} \subset \mathbb{R}^2 \times [0, T]$ of border $\partial \mathbf{V}$ and of outward unit normal vector \mathbf{n} . Then, let us integrate System (2) on \mathbf{V} and apply the Green formula:

$$\partial_t \int_{\mathbf{V}} \mathbf{w} \, dx + \int_{\partial \mathbf{V}} \mathbf{f}(\mathbf{w}) \cdot \mathbf{n} \, dv = \int_{\mathbf{V}} \mathbf{G}(x, \mathbf{W}) \, dx.$$

The Jacobian matrix in the primitive variable of the flux across the normal \mathbf{n} is then defined as

$$Jac = \left(\frac{\partial \mathbf{w}}{\partial \mathbf{W}} \right)^{-1} \frac{\partial \mathbf{f} \cdot \mathbf{n}}{\partial \mathbf{w}} \left(\frac{\partial \mathbf{w}}{\partial \mathbf{W}} \right)$$

where

$$\frac{\partial \mathbf{f} \cdot \mathbf{n}}{\partial \mathbf{w}} = \begin{pmatrix} 0 & n_1 & n_2 \\ (gh - u_1^2)n_1 - u_1 u_2 n_2 & 2u_1 n_1 + u_2 n_2 & u_1 n_2 \\ -u_1 u_2 n_1 + (gh - u_2^2)n_1 & u_2 n_1 & u_1 n_1 + 2u_2 n_2 \end{pmatrix}$$

so that λ_i is the i -th eigenvalue and r_i is the i -th right eigenvector of the matrix Jac :

$$Jac = \begin{pmatrix} 0 & n_1 & n_2 \\ (gh - u_1^2)n_1 - u_1 u_2 n_2 & 2u_1 n_1 + u_2 n_2 & u_1 n_2 \\ -u_1 u_2 n_1 + (gh - u_2^2)n_1 & u_2 n_1 & u_1 n_1 + 2u_2 n_2 \end{pmatrix}.$$

The item 2. and 3. are obtained through a simple calculation. Finally, for smooth solutions, the entropy equality (6) is obtained as the results of of $h \times (4) \cdot \mathbf{u} + \left(\partial_t h + \operatorname{div}(h\mathbf{u}) \right) (gh + gZ)$ which ends the proof. \square

Remark 1. We recall that the solutions of the above System (2) can exhibit in finite time discontinuities, corresponding to hydraulic jumps or bores, even if the initial data $\mathbf{w}(x, 0) = \mathbf{w}_0(x)$ is smooth. It can be recovered by completing System (2) with the entropy inequality (6). Even if we are not able to prove the uniqueness in the multi dimensional case, this inequality allows to select the physical relevant solution and provide a “smoothness” indicator since the entropy satisfies a conservation equation only in regions where the solution is smooth and an inequality when the solution develops discontinuities. Thus, the discrete quantity S , called *the numerical density of entropy production*, can be considered as a measure of the amount of violation of the entropy equation (as pointed out in [4, 16]). Therefore it can be used as a suitable *mesh refinement criterion* as an ”error indicator”, see Sect. 3 and [8].

2.2 Finite volume approximation

The computational domain $\Omega \subset \mathbb{R}^d$ is split into a set of control volumes, also referred as cells, $\Omega = \cup_k C_k$ of mesh size $|C_k|$. The source term is upwinded at the cell interfaces through an hydrostatic reconstruction [2] to obtain a well-balanced scheme.

On a given cell C_k , noting $w_k(t)$

$$\mathbf{w}_k(t) \simeq \frac{1}{|C_k|} \int_{C_k} \mathbf{w}(x, t) \, d\mathbf{x}$$

the approximation of the mean value of the unknown $\mathbf{w}(x, t)$ on C_k at time t , and integrating (2) over each cell, we obtain:

$$\int_{C_k} \frac{\partial \mathbf{w}(t)}{\partial t} + \sum_a \int_{\partial C_k/a} \mathbf{f}(\mathbf{w}(t)) \cdot \mathbf{n}_{k/a} \, ds = 0$$

where $\mathbf{n}_{k/a}$ denotes the unit normal vector on the boundary $\partial C_{k/a}$ between cells k and a .

Next, $F(\mathbf{w}_k(t), \mathbf{w}_a(t), \mathbf{n}_{k/a})$ the flux approximation being written

$$\mathbf{F}(\mathbf{w}_k(t), \mathbf{w}_a(t), \mathbf{n}_{k/a}) \approx \int_{\partial C_{k/a}} \mathbf{f}(\mathbf{w}(t)) \cdot \mathbf{n}_{k/a} ds,$$

the semi-discrete finite volume approximation of Eqs. (2) (see for instance [12, 27, 9]) is obtained:

$$\frac{\partial \mathbf{w}_k(t)}{\partial t} + \frac{1}{|C_k|} \sum_a \mathbf{F}(\mathbf{w}_k(t), \mathbf{w}_a(t), \mathbf{n}_{k/a}) = 0 \quad (7)$$

where $\mathbf{F}(\mathbf{w}_k(t), \mathbf{w}_a(t), \mathbf{n}_{k/a})$ is defined via the Godunov solver, i.e. it is computed with the exact solution of the 1D Riemann problem at the interface k/a with the states $\mathbf{w}_k(t)$ and $\mathbf{w}_a(t)$ (for further details see, for instance, [27]).

The source term is upwinded at the cell interface following the hydrostatic reconstruction. It means that for each interface, the numerical flux

$$\mathbf{F}(\mathbf{w}_k(t), \mathbf{w}_a(t), \mathbf{n}_{k/a})$$

is replaced by the flux at left hand side $\mathbf{F}_k(\mathbf{w}_k(t), \mathbf{w}_a(t), \mathbf{n}_{k/a}, \Delta Z_{k/a})$ and the right hand side $\mathbf{F}_a(\mathbf{w}_k(t), \mathbf{w}_a(t), \mathbf{n}_{k/a}, \Delta Z_{k/a})$ of the interface k/a . These new fluxes are constructed as follows

$$\begin{aligned} \mathbf{F}_k(\mathbf{w}_k(t), \mathbf{w}_a(t), \mathbf{n}_{k/a}, \Delta Z_{k/a}) &= \mathbf{F}(\mathbf{w}_k^*, \mathbf{w}_a^*, \mathbf{n}_{k/a}) + \begin{pmatrix} 0 \\ \frac{g}{2}(h_k^2 - (h_k^*)^2)n_1 \\ \frac{g}{2}(h_k^2 - (h_k^*)^2)n_2 \end{pmatrix} \\ \mathbf{F}_a(\mathbf{w}_k(t), \mathbf{w}_a(t), \mathbf{n}_{k/a}, \Delta Z_{k/a}) &= \mathbf{F}(\mathbf{w}_k^*, \mathbf{w}_a^*, \mathbf{n}_{k/a}) + \begin{pmatrix} 0 \\ \frac{g}{2}(h_a^2 - (h_a^*)^2)n_1 \\ \frac{g}{2}(h_a^2 - (h_a^*)^2)n_2 \end{pmatrix} \end{aligned}$$

by means of reconstructed states (satisfying the still water steady state equation (5))

$$\begin{aligned} \mathbf{w}_k^* &= (h_k^*, \mathbf{u}_k), \\ \mathbf{w}_a^* &= (h_a^*, \mathbf{u}_a), \\ h_k^* &= \max(0, h_k - \max(0, \Delta Z_{k/a})), \\ h_a^* &= \max(0, h_a - \max(0, -\Delta Z_{k/a})). \end{aligned}$$

In these formula, $\Delta Z_{k/a}$ stands for the jump of Z across the interface k/a . The scheme is therefore well-balanced by construction, i.e., the still water steady states are exactly satisfied.

Finally, Equations (2) are completed with the entropy inequality (6). Following Ersoy et al. [13, 8], the entropy inequality (6) is approximated using the semi-discrete finite volume scheme (7). The obtained discrete quantity, called the *numerical density of entropy production*, is then used as a *mesh refinement criterion* (see

Remark 1, Sect. 3 and [8, 22]). For further details on the construction of the numerical scheme we refer to [8] since the definition of the numerical fluxes reduces to a one dimensional computational at each interface k/a .

3 BB-AMR method

Defining a robust mesh refinement criterion for multi dimensional configurations is not enough to design a suitable numerical solver. The mesh refinement threshold and the treatment of data are also crucial points and in particular the way to distribute refined cells and to share the memory in a parallel process. This point is handled within an Automatic Mesh Refinement Threshold framework in a hierarchical Block-Based way called *parameterless* BB-AMR. The global strategy to adapt the mesh and how to manage data are presented.

3.1 AMR and Automatic thresholding methods

The efficiency of the numerical density of entropy production (see Remark 1) as a relevant mesh refinement parameter have been already demonstrated in a previous work [8]. It has been numerically observed (and from theoretical considerations) that the production of the numerical density of entropy is almost zero for smooth solution and non-positive when the solution develops discontinuities. As a consequence, the mesh is automatically and proportionally (with respect to the production) refined inside area where the production is non zero. More precisely, Ersoy et al. [8] have demonstrated that, for the one dimensional gas dynamics equation, the support of the relative error coincides with the support of the numerical density of entropy production. The extension toward the multi dimensional case is detailed in the case of two-fluid flows in [14]. Therefore, according to the finite volume approximation defined in Sect. 2.2, a local numerical entropy production S_k^n is computed with absolute value on each cell at time t_n and compared to a mesh refinement threshold $\alpha > 0$. Finally, for each cell C_k :

- if $S_k^n > \alpha$, the mesh is refined and,
- if $S_k^n < \alpha$ the mesh is coarsened.

In the literature, the threshold parameter α is determined according to the simulated case requirements, to reach a relevant compromise between computational cost and accuracy. Moreover, the smaller α is, the more accurate the results are at the expense of CPU time and yields to unnecessary refinement. To obtain a suitable balance between computational cost and accuracy, following [22], we set automatically the threshold $\alpha = \alpha_{\text{PE}}$ such that

$$\alpha_{\text{PE}}d(\alpha_{\text{PE}}) = \max_{0 < \alpha \leq S_m} (\alpha d(\alpha))$$

where d is the decreasing rearrangement (also known as the distribution) function of the mesh refinement criterion S (here the absolute value of the numerical density of entropy production) and S_m its mean value. More precisely, for each threshold $\alpha \in (0, S_m]$, $d(\alpha)$ is the Lebesgue measure of the set $d(\alpha) = \{S(x) > \alpha\}$ and provides a complete description of the criterion in terms of local local maxima which are sorted canonically (from the smallest to the largest). In practice, the threshold α_{PE} verifies the desired behavior, namely $\alpha_{\text{PE}} < S_m$ catching the smallest local maxima for discontinuous flows and $\alpha_{\text{PE}} \approx S_m$ for smooth flows. As a consequence, it allows to reach a relevant compromise between computational cost and accuracy by indicating efficiently the cells to be refined. Therefore, *the presented AMR process, in comparisons with existing methods, is now parameterless.*

From a numerical viewpoint, the distribution function is defined as follows. Let us consider a given discrete mesh refinement criterion $S(x, t_n) = \sum_{k=1}^N S_k^n \mathbb{1}_{C_k}(x)$ where N is the total number of cells at time t_n and $\mathbb{1}_{C_k}(x)$ is the indicator function of the set C_k . Let $(\alpha_j)_{0 \leq j \leq M}$ be an increasing sequence of $M + 1$ threshold parameter such that $(\alpha_j)_{0 \leq j \leq M} = \left(S_m \left(\frac{j}{M} \right)^\beta \right)_{0 \leq j \leq M}$. Then, the distribution function $d_j = d(\alpha_j)$ is given by

$$d_j = \#\{k ; S_k^n > \alpha_j\}$$

where $\#$ is the number of elements in the set $\{S_k^n > \alpha_j\}$. In [22, Section 4], it is shown that the method is quite insensitive to the parameter β , set to 2 here. For all applications M is fixed to $M = 1000$ points. Interested readers can found details in [22].

3.2 Data management

For the one dimensional case, the local mesh refinement procedure is constructed following dyadic tree applied at each time step. ‘‘Macro cells’’ are used to be easily refined by generating hierarchical grids. Each cell can be split in two. Dyadic cells graph are thus produced, in basis 2 numbering, to allow a quick computing scan to determine the adjacent cells. For stability reasons, the mesh refinement level cannot exceed 2 between two adjacent cells. More details can be found in [8].

The multi dimensional extension of the mesh refinement procedure is a difficult task. Interesting works have been presented for 2D Cartesian grid or quad-tree [3, 29, 21, 30], octree for 3D simulations [18, 10], and anisotropic AMR [6, 15]. For the Saint-Venant system (1) the extension from 1D to 2D leads naturally to quad-tree meshing. But, the presence of a complex moving interface (composed of rarefaction and shocks) implies to re-mesh at each time step, which is obviously a costly process. Guided by the need to reach a relevant compromise between the contradictory aims of solution accuracy and computing speed, a Cartesian block-based mesh approach is introduced, somehow like in [28, 31]. If the mesh is not refine at

each time step, the patch where the grid should be refined must be enlarged. The grid generation in the framework of BB-AMR can be developed as follow.

- Firstly, the initial computational domain is divided in several hexahedral elements, called blocks. For the sake of simplicity, Fig. 1(a) represents a regular quadrilateral mesh.
- Secondly, we define for each block a discretization n_x in x-direction, n_y in y-direction and a level of refinement l_r such that the total number of cell in x-direction of the block will be $2^{l_r}n_x$. As the mesh refinement level cannot exceed 2 between two adjacent blocks, the level of mesh refinement is automatically adapted if necessary (as for example for the blocks B_5 and B_2 in Fig. 1(b))
- Thirdly, in order to balance the CPUs load, the cells of each block are re-distributed in a fixed number of domains according to the Cuthill-McKee numbering, see Fig. 1(c).
- Finally, the unstructured mesh is built for each domain.

For each refined cells (or blocks), averaged values are projected on each sub-cell and fluxes are computed as simply as possible to avoid heavy computation. At the interface between two domains, ghost cells are created in order to transfer the necessary data to the explicit time integration scheme at each time step. The number of domain being fixed, each domain are loaded in a given MPI process. These MPI processes are then load on a fixed number of cores (not necessary the same).

The re-numbering and re-meshing being expensive, the mesh is finally kept constant on a time interval, called AMR time-step, given by the smallest block (rather by the smallest cell) and the maximum velocity. Details on the BB-AMR are given in [14, 1].

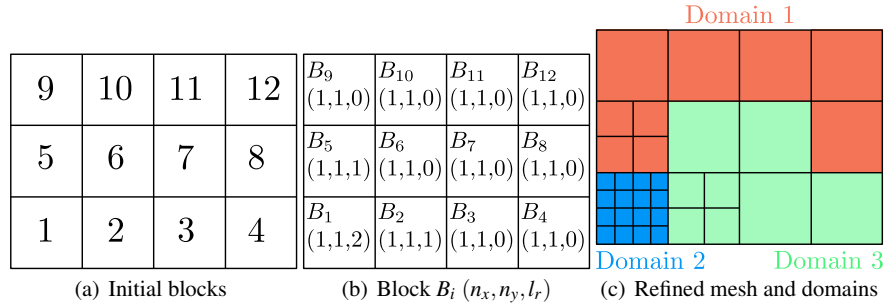


Fig. 1 Illustration of the BB-AMR process

4 Real-life Applications

We now focus on the overall performance of the BB-AMR scheme with automatic thresholding confronted to experimental and state-of-the-art numerical results¹. For each test case, the numerical simulations are carried out on adaptive grids (using the BB-AMR method previously presented) and on uniform grids. The comparisons of both simulations allow to show the reliability and the efficiency of the parameterless BB-AMR method. To simplify the analysis, we have considered only 1 domain, i.e., 1 processor. Interested readers can find a detailed numerical study with several domains in [14]. Moreover, unless otherwise indicated, the expression "numerical density of entropy production" is referred to the absolute value of the numerical density of entropy production.

4.1 Solitary wave propagation over a two dimensional reef

This benchmark aims at reproducing a set of laboratory experiments carried out at the O.H. Hinsdale Wave Research Laboratory, Oregon State University (OSU, see Roeber *et al.* [26] and Roeber and Cheung [25]). These experiments involve the propagation, run-up, splash-up and reflection of high amplitude solitary waves on two-dimensional reefs. Their purpose is on one hand to investigate processes related to breaking, bore formation, dispersion, and passage from sub- to super-critical flows, and on the other hand, to provide data for the validation of near-shore wave models in fringing reef. Even if the Saint-Venant model (2) is not relevant for dispersive flows, through this test case we show that globally experimental data are well-captured and the cells are well-refined thanks to the automatic thresholding technique.

4.1.1 Experimental settings

The geometry of the test considered here is shown in Fig. 2. The length of the basin is of 104 m, however the computational domain is delimited by a reflecting wall placed at $x = 83.7$ m. The reef starts at $x = 25.9$ m with a nominal slope of $1/12$. The actual slope is such that the height of 2.36 m is reached after $x = 28.25$ m. At this station a 0.2 m height crest is mounted. The offshore slope of the crest is the same of the reef, and the length of its plateau is of 1.25 m. The on-shore side has a slope of $1/15$ giving a nominal length for the crest basis of 6.65 m (using the actual offshore slope, a crest basis of 6.644 m is obtained). For the computations, the use of the nominal slope values is prescribed. This gives an offshore length of the crest slope (starting at $x = 28.25$ m) of 2.4 m. Thus, the bathymetry is

¹ The numerical soft used for the following test cases is the EOLENS code developed by the Institut de Mathématiques de Toulon (IMATH) and Principia.

$$Z(x) = \begin{cases} -2.5 & \text{if } x < 25.9, \\ \frac{(x-25.9)}{30.8} 2.565 - 2.5 & \text{if } 25.9 < x < 56.7, \\ 0.065 & \text{if } 56.7 < x < 57.65, \\ \frac{(57.95-x)}{3} 0.2 + 0.0650 & \text{if } 57.65 < x < 60.95, \\ -0.1360 & \text{if } x > 60.95. \end{cases}$$

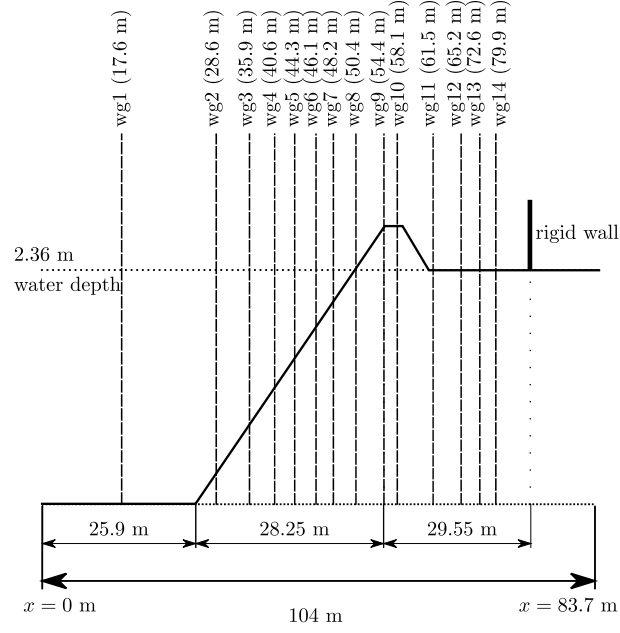


Fig. 2 Experimental settings and wave gauges locations

The initial depth at still water is $h_0 = 2.5$ m, yielding to a partially submerged crest, and a depth behind it (on-shore side) of 0.14 m. The initial solution consists of a solitary wave of amplitude $A = 0.75$ m providing a non linearity ratio of $\frac{A}{h_0} = 0.3$. For numerical purpose, the initial data consist of a solitary wave centered at $x_0 = 17.6$ m of amplitude

$$\eta(x, 0) := h(x, 0) + Z(x) = \frac{A}{\cosh\left(\sqrt{\frac{3A}{4h_0^3}}(x - x_0)\right)^2} \quad (8)$$

and velocity

$$u(x, 0) = \sqrt{g(h_0 + A)}\eta(0, x)/h_0 \quad (9)$$

which corresponds to the experimental data at the dimensionless time $\tilde{t} = t \sqrt{\frac{g}{h_0}} = 55.03$ as displayed in Fig. 3(a).

4.1.2 Numerical results vs experimental results

Figures 3(a) and 3(b) shows the propagation of the solitary wave over the slope starting at $x = 25.9$ m. Figure 3(c) represents the surface elevation at the dimensionless time $\tilde{t} \approx 69$ where the wave develops into a plunging breaker onto the reef crest². From an experimental viewpoint, $\tilde{t} \approx 69$ corresponds to the subsequent time of overturning of the free surface. In Fig. 3(d), the wave hitting the free surface creates a downstream and upstream propagating bore as shown in Fig. 3(e). The downstream propagating bore is reflected at the end wall at $x = 83.7$ m and propagates in opposite direction as displayed in Fig. 3(f).

² The authors would thanks V. Roeber to furnish the experimental data for this test case.

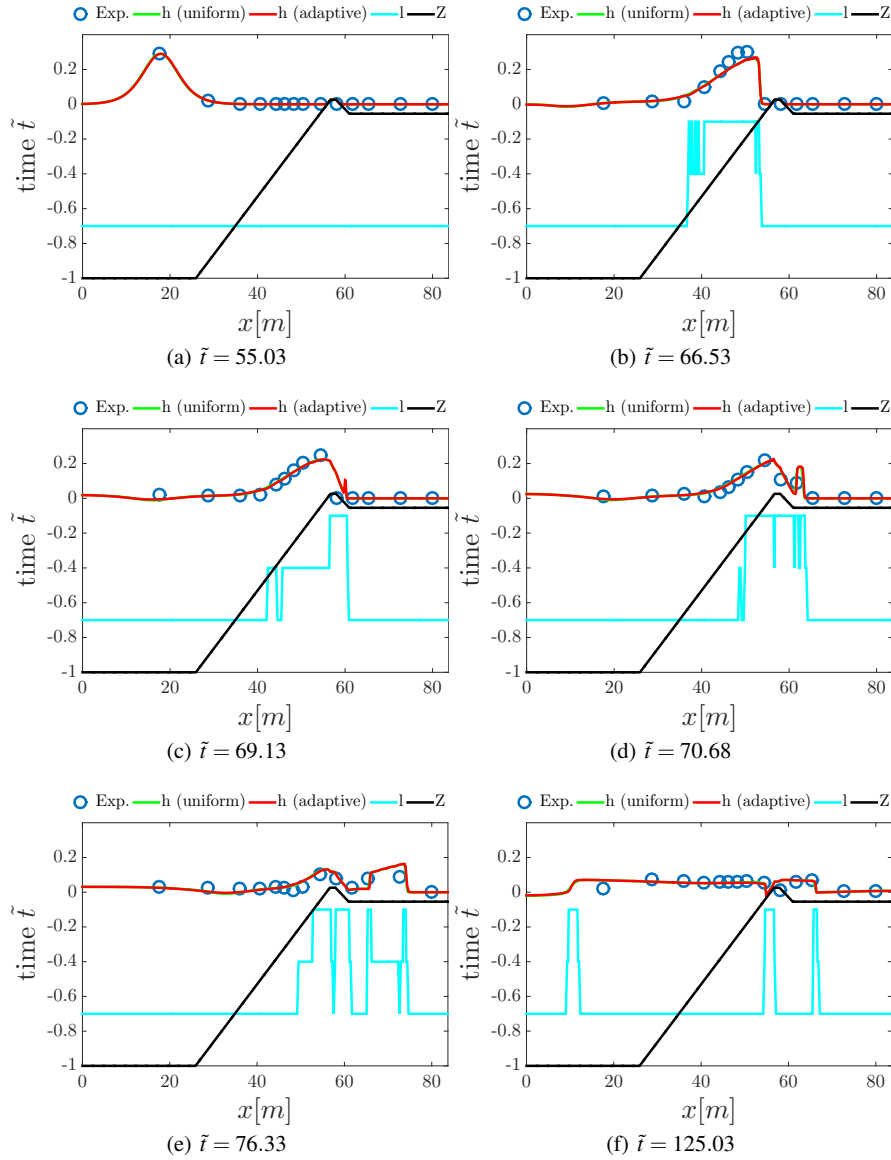


Fig. 3 Surface profiles of solitary wave propagation over an exposed reef crest. Confrontation of experimental data (blue circles) to numerical data computed on a uniform grid (solid green line) and on an adaptive grid (solid red lines). The solid cyan line represents the mesh level and the black one the bathymetry

In Figs. 3(a)–3(f), the experimental surface elevation (represented with blue circles) over the bathymetry Z for several dimensionless time \tilde{t} are compared to the

obtained numerical results (green and red lines-circles)³. The numerical solution is computed on a uniform grid (green lines-circles) composed of 1000 cells and on an adaptive grid (red lines-circles) initially composed of 200 cells/blocks with $l_{\max} = 4$ where l_{\max} is the maximum level of the mesh refinement. For both numerical experiments, the CFL number is set to 0.99. Free boundary condition at $x = 0$ m and reflecting boundary condition at $x = 83.7$ m are imposed. Table 1 summarizes the numerical parameters used for this test case.

	Adaptive mesh simulation	Uniform mesh simulation
Simulation time	240	240
Number of cells	200-560	1000
Re-meshing time step	0.05 s	not applicable
Time order integration	1	1
Space order integration	1	1
CFL	0.99	0.99

Table 1 Numerical parameters

The obtained numerical results, as shown in Figs. 3(a)–3(f), are rather well-computed when compared to experimental data. However, the wave height is slightly underestimated (see Fig. 3(c)) while the hydraulic bore propagation front height is overestimated (see Fig. 3(d)).

We now compare in Figs. 4(a)–4(f), the computed and the experimental water elevation at the wave gauges displayed in Fig. 2. The oscillating shape of the experimental measures almost corresponds to dispersive effects which cannot be detected with the Saint-Venant model (2). However, the shape of the numerical results almost coincides with the experimental data for wave gauges located before $x = 54.4$ m as shown in Figs. 4(a)–4(c). The point $x = 54.4$ m is located just before the crest and in particular it corresponds to the region where the overturning of the free surface is experimentally observed. It yields numerically to a small shift as observed in Figs. 4(d)–4(f).

³ Intel(R) Core(TM) i5-2500 CPU @ 3.30GHz

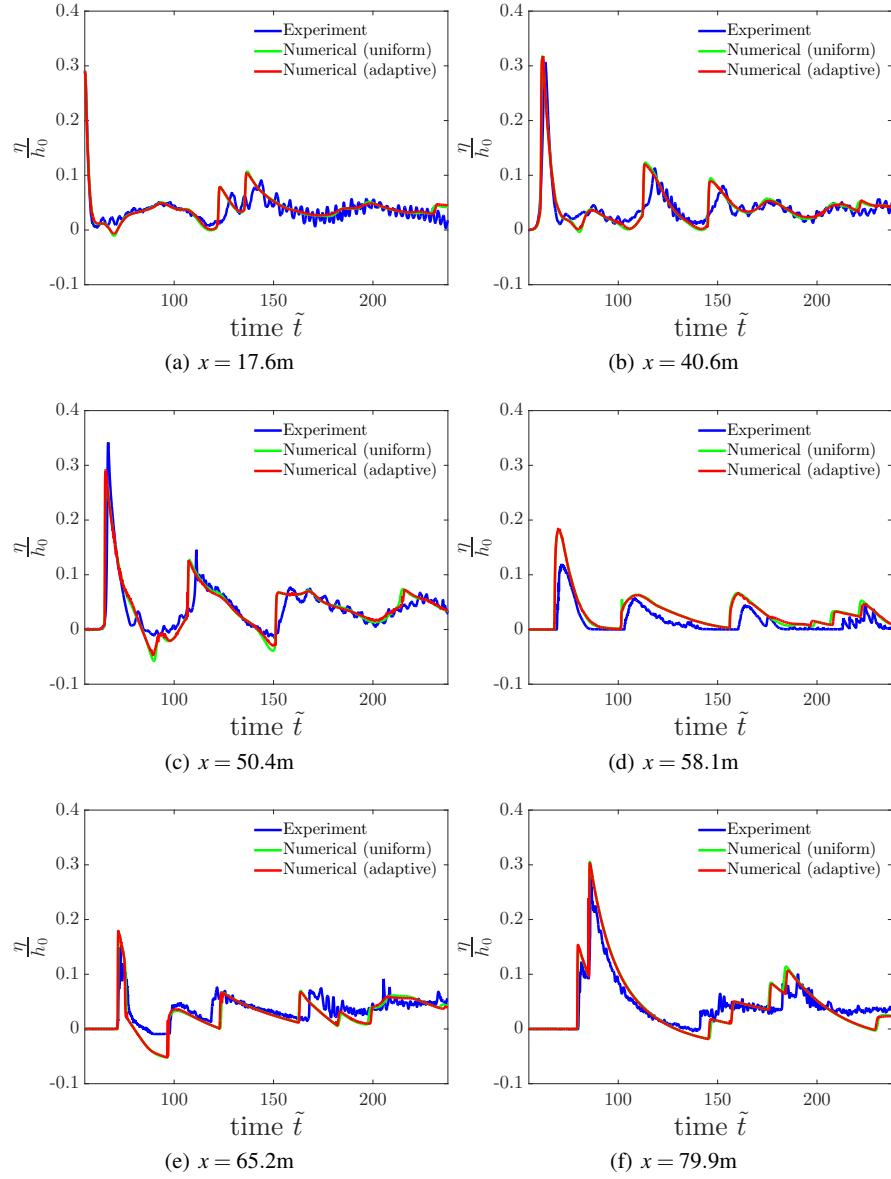


Fig. 4 Surface profiles of solitary wave propagation in time at wave gauges 1 to 6. Confrontation of experimental data (solid blue line) to numerical data computed on a uniform grid (solid green line) and on an adaptive grid (solid red line)

4.1.3 AMR performance

We now compare the performance of the adaptive and uniform method. The adaptive scheme uses an average of 356 cells against 1000 cells (see Fig. 5(b)) for the uniform one and the CPU-time is 95 s against 210 s.

We observe that the computation on adaptive grid is comparable to the numerical solution on uniform grid and computed about 2 times faster with less cells. We see also in Figs. 3(a)-3(f) that for each time step, the threshold parameter is automatically well-set to capture efficiently the region to refine (see also Fig. 5(a)). During the first 50 non-dimensional time, we have already pointed out that the solitary wave propagates and splash-up. This phenomenon is characterized by a steep gradient flow followed by a discontinuous flow for the Saint-Venant system, see Fig 3. In view of the remarks in Sect. 3.1, we observe that for almost all $\tilde{t} \in (55, 100)$, the threshold is set to $\alpha_{\text{PE}} < S_m$. For $\tilde{t} > 100$, the flow is almost smooth and therefore $\alpha_{\text{PE}} \approx S_m$. It is also interesting to highlight that only the region of interest are automatically refined as shown in Fig. 3(e). Therefore, we recover the desired behavior predicted by Pons and Ersoy [22, Section 4]. Moreover, in view of the confrontation with the computation on uniform grid, this test shows the reliability/efficiency of the automatic selection of the threshold for multiple scale flows.

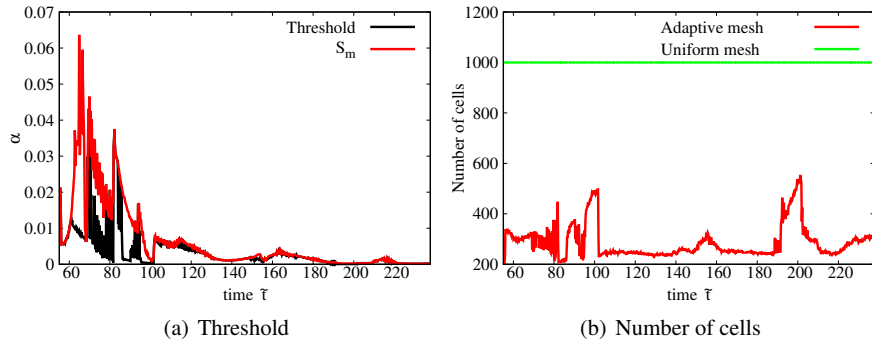


Fig. 5 Time evolution of the mesh refinement threshold and the number of cells

4.2 Solitary wave propagation over an irregular three-dimensional shallow shelf

In this test case, the numerical simulation of a solitary wave propagation over a complex three dimensional bathymetry is performed. This test case was experimentally introduced in [19] to understand the turbulence and kinematic properties associated with a breaking solitary wave. In this experiment, the propagation, run-up, drying and flooding phenomena are involved.

4.2.1 Experimental settings

A laboratory experiment was conducted in a large wave basin which was 48.8 m long and 26.5 m wide at Oregon State University for which free surface elevations and fluid velocities were recorded at several locations by wave gauges (WG) and velocity captors (ADV). A single solitary wave of height of 39 cm at $x = 5$ m is produced with a piston-type wave-maker over a complex bathymetry illustrated in Fig. 6. The bathymetry begins with a flat part allowing to generate properly a solitary wave. Then, starting at $x = 10.2$ m of the wave-maker and extending to $x = 25.5$ m, a complex three dimensional shelf is built. The associated slope to this zone is variable but less and less steep. Beyond that, a very small one dimensional slope finishing on a flat onshore zone is built. The experimental basin is closed by walls. Fluid height and velocities are recorded during 45 s. Finally, the material used for the walls and the bathymetry is made of smooth concrete so that one can assume almost a frictionless flow. For further informations on this experiment, interested readers can found more details in [19].

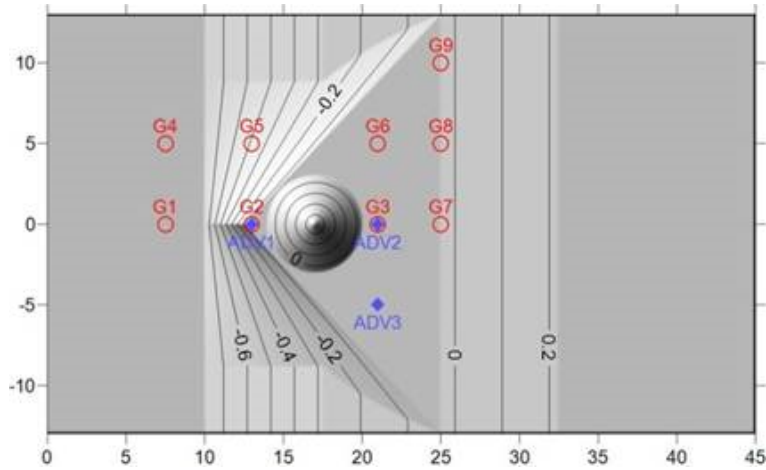


Fig. 6 Experimental settings

4.2.2 Numerical vs experimental results

The domain is numerically extended to $x = -5$ m with a water depth of 78 cm to impose a solitary wave instead of reproducing the wave maker movement. The simulated solitary wave is a first order solution of the Boussinesq equation (see Eqs. (8) and (9)) with $x_0 = 5$ m and $\frac{A}{h_0} = 0.5$. For computational purpose, we have considered 128 initial blocks composed of 7 500 cells for the initialization of the adaptive computation and almost 33 000 cells for uniform mesh computation. For both simulations, the CFL number is set to 0.5. Reflecting boundary conditions are

prescribed to walls. Table 2 summarizes the numerical parameters used for this test case.

	Adaptive mesh simulation	Uniform mesh simulation
Simulation time	30 s	30 s
Number of blocks	128	128
Number of cells	7 500-25 000	33 000
Re-meshing time step	0.25 s	not applicable
Time order integration	2	2
Space order integration	2	2
CFL	0.5	0.5

Table 2 Numerical parameters

To illustrate the main propagation phenomena, we have displayed the numerical solutions calculated at time $t = 0.5$ s (see Fig. 7(a)), $t = 2.5$ s (see Fig. 7(b)), $t = 5.75$ s (see Fig. 7(c)) and $t = 23.75$ s (see Fig. 7(d)). The Fig. 7(a) represents the profile of the solitary wave which propagates towards the coast. Figures 7(b) and 7(c) show respectively the wave hitting the conic island and then the beginning of the flooding. Figure 7(d) shows the flow after the wave run-up and its reflection on the right side of the domain.

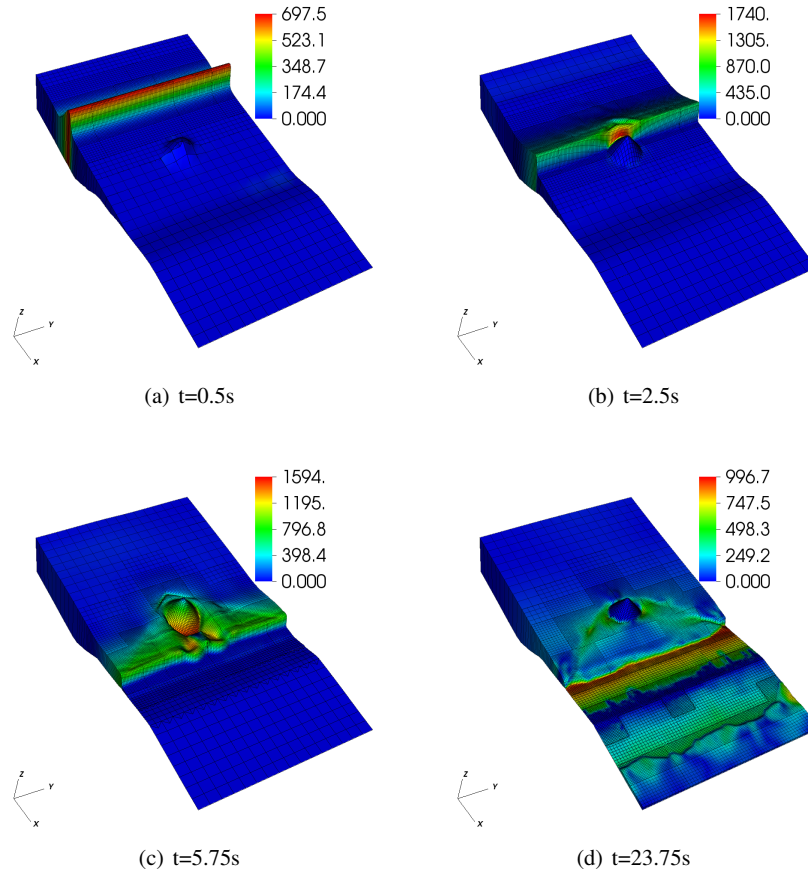


Fig. 7 Numerical water height (coloration is issue from the kinetic energy)

The global numerical results are now confronted to the experimental data recorded at the waves gauges 2, 4, 6 and 7 (WG) for the water height, see Fig. 6 and 8, and to the velocity captors 2 and 3 (ADV) for the first u component of the velocity vector, see Fig. 6 and 9. For the wave gauge 2 to 7 (see Figs. 8(a)–8(d)), the global shape of the free surface is in quite good agreement with the experimental data but some differences are observed. The numerical solution computed with the Saint-Venant system (2) slightly underestimates the water level at the wave gauge 2 (see Fig. 8(a)) and overestimates at the wave gauge 4 (see Fig. 8(b)). Moreover, the numerical solution is rather accurate up to time $t = 15$ s at the wave gauges 6 and 7 (see Fig. 8(c) and 8(d)). For $t \geq 15$ s, the dispersive effects, not reproduced with the Saint-Venant system, begin significant and induce slightly different results between experimental and numerical data. Similar conclusion can be drawn up for the first component of the vector velocity on the captor 2 (see Fig. 9(a)) and 3 (see Fig. 9(b)). Globally,

the numerical results on this test case are in a good agreement and can be improved with a dispersive model.

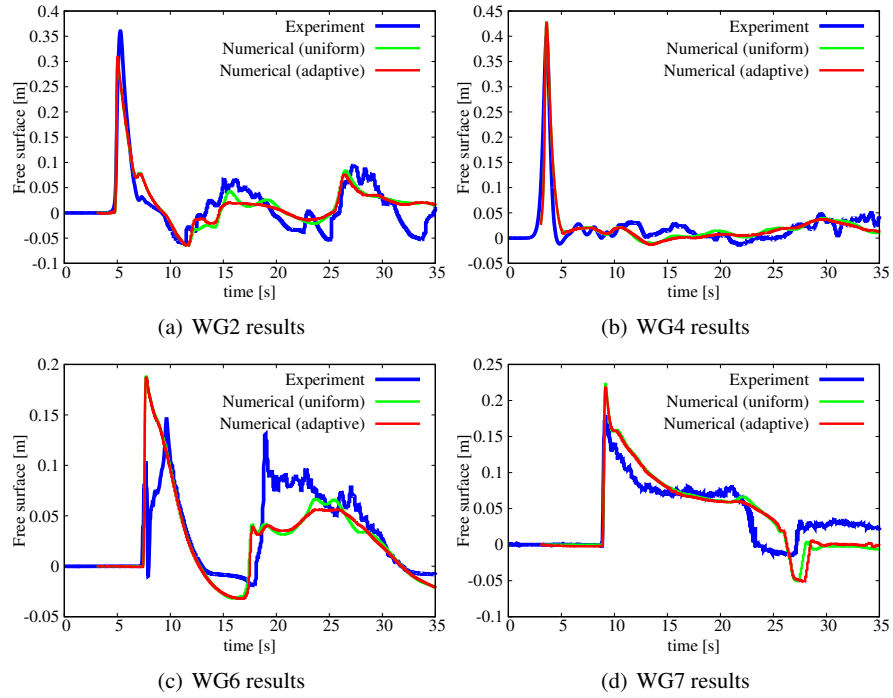


Fig. 8 Free surface results at different positions : experimental data versus numerical simulation with and without mesh adaptivity

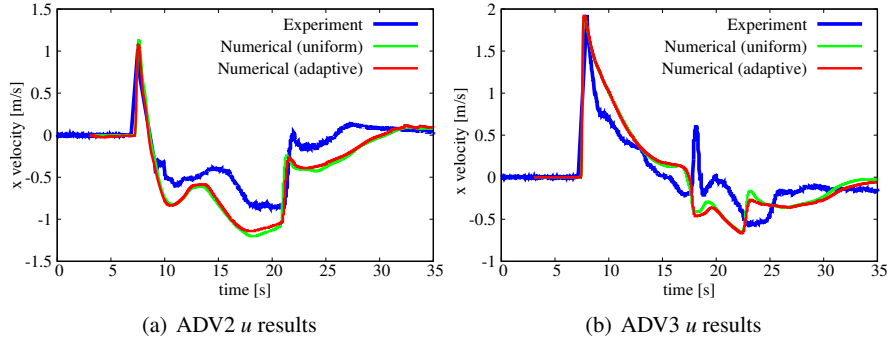


Fig. 9 Velocity results at different positions : experimental data versus numerical simulation with and without mesh adaptivity

4.2.3 AMR performance

To highlight the efficiency of the mesh refinement parameter and the automatic selection of the threshold, the numerical density of entropy production (without absolute value) and the associated mesh are displayed at several times in Fig. 10, (time $t = 0.5$ s in Fig. 10(a), for $t = 2.5$ s in Fig. 10(b), for $t = 5.75$ s in Fig. 10(c) and for $t = 23.75$ s in Fig. 10(d)). The time evolution of the mesh refinement threshold selecting the region to be refined is plotted in Fig. 11(a). It can be noticed that according to the complex flow zones the numerical density of entropy production indicates which areas need to be refined. Secondly, the mesh density is well distributed on these zones which illustrate the efficiency of the automatic thresholding method.

For instance, in Fig. 7(c), the region which needs to be refined is clearly the zone around the "conic island". According to Fig. 10(c), the numerical density of entropy production in this area shows that the local maxima (lowest value without the absolute value) are of almost of order 0.005 and clearly follows the wave front. We see in Fig. 11(a), approximately at time $t = 5.75$ s, that the threshold is automatically fixed to almost 0.001 and therefore allows to perform a suitable refinement in the region of interests as observed in Fig. 7(c) or 10(c).

Compared to the computation on a uniform grid (see Figs. 8 and 9), the adaptive mesh refinement scheme requires only an average of 13000 cells against 33000 cells (see Fig. 11(b) for the number of cells for all the simulation). For the same accuracy, the adaptive scheme allows to save almost 60% of cells with respect to the uniform mesh. During the adaptive simulation, the maximum refined area is reached around the time $t = 12$ s which corresponds to the time where the free surface oscillates behind the island and the wave run-up on the "coast". This result makes sense because the mesh has to follow a lot of oscillation and not only a solitary wave. Globally, all these cells saved allow to speed up the computation by 2.5 time for this test case.

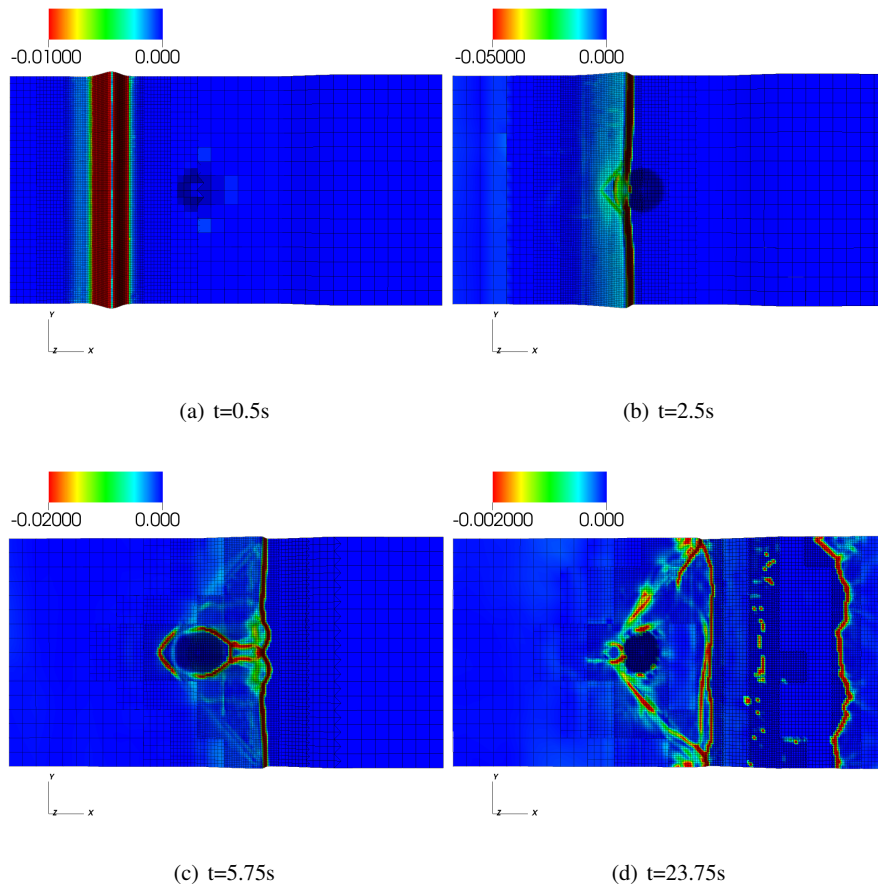


Fig. 10 Adaptive mesh (coloration is issue from the numerical density of entropy production without absolute value)

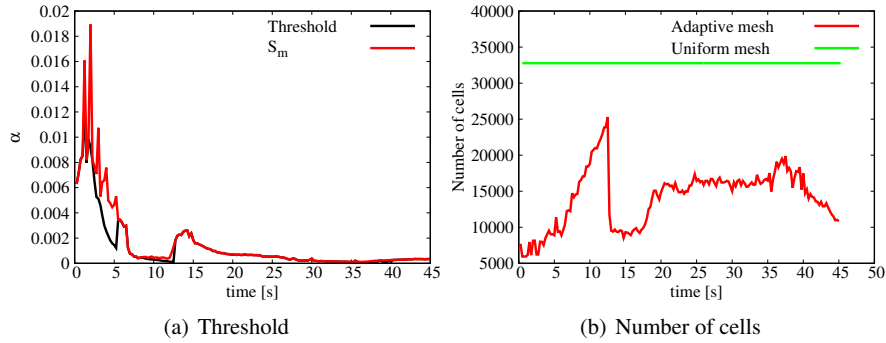


Fig. 11 Time evolution of the mesh refinement threshold and the number of cells

4.3 Tsunami runup onto a complex three dimensional Monai-Walley beach

This test case concerns the Hokkaido-Nansei-Oki tsunami, in 1993, that struck the Okushiri Island in Japan. The tsunami run-up records was about 30 m height and the currents speed of order 10-18 m/s for which relevant high-quality data were saved.⁴ The largest value run-up (32 m) was recorded near the Monai-Walley beach.

4.3.1 Experimental settings

To understand this complex run-up, a 1/400 scale laboratory model of Monai was realized in a large-scale tank of 205 m long, 6 m deep, and 3.5 m wide at the Central Research Institute for Electric Power Industry (CRIEPI) in Abiko (Japan). The offshore incident wave on a water depth $d = 13.5$ cm is prescribed. There are reflective vertical side-walls at $y = 0$ and 3.5 m as for the x boundaries. The bathymetry as well as the coastal topography reproduced in the laboratory experiment are represented⁵ in Fig. 12(b) and 12(c).

The input wave at $x = 0$ m is a leading-depression height of -2.5 mm with a crest of 1.6 cm, as displayed in Fig. 12(a). In the experiment, the waves are measured at thirteen locations and complete time histories are given at three locations, $(x, y) = (4.521, 1.196)$, $(4.521, 1.696)$, and $(4.521, 2.196)$ in meters, see also [17]. In

⁴ Several sources and data can be found, see for instance http://nctr.pmel.noaa.gov/benchmark/Laboratory/Laboratory_MonaiValley/ or http://isec.nacse.org/workshop/2004_cornell/bmark2.html

⁵ sources and pictures are available at http://isec.nacse.org/workshop/2004_cornell/bmark2.html

contrast with the previous experimental test cases, the dispersive phenomena can be neglected here.

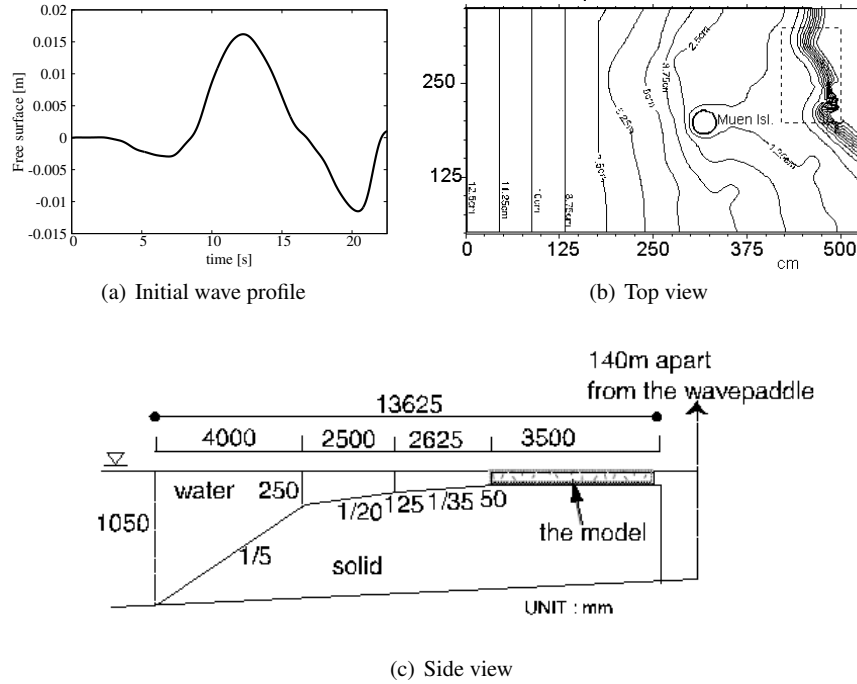


Fig. 12 Experimental settings

4.3.2 Numerical vs experimental results.

The initial condition for this test case is the lake at rest and an imposed water height on the left side wall, see Fig. 12(a). Wave gauges data ⁶ experimental results are given up to time $t = 100$ s. However, the information concerning the input wave profile (see Fig. 12(a)) at $x = 0$ m is only available up to time $t = 22$ s. For numerical purpose, we prescribe for $t > 22$ s, a constant free surface level $\eta = 0$ m on the left wall. In practice, due to this assumption, some artificial reflecting waves can occur on the wave-maker beyond $t > 22$ s and can reach the free surface gauges around $t = 30$ s. Finally, to avoid any wrong comparisons due to these artificial reflections the comparisons between the experimental gauges and the numerical results will be only done over the first 30 s of the experiment.

⁶ Wave gauges data are available at the address http://nctr.pmel.noaa.gov/benchmark/Laboratory/Laboratory_MonaiValley/

For computational purpose, we have considered 240 initial blocks composed of 8 000 cells for the initialisation of the adaptive simulation and 62 000 cells for uniform mesh simulation. For both simulations, the CFL number is set to 0.5. Reflecting boundary conditions are prescribed to walls. Table 3 summarizes the numerical parameters used for this test case.

	Adaptive mesh simulation	Uniform mesh simulation
Simulation time	30 s	30 s
Number of blocks	240	240
Number of cells	8 000-40 000	62 000
Re-meshing time step	0.25 s	not applicable
Time order integration	2	2
Space order integration	1	1
CFL	0.5	0.5

Table 3 Numerical parameters

To illustrate this test case we show in Fig. 13, the propagation of the wave (colored with the kinetic energy) at time $t = 11.25$ s (Fig. 13(a)), $t = 13.25$ s (Fig. 13(b)), $t = 16$ s (Fig. 13(c)) and $t = 17.5$ s (Fig. 13(d)). The Fig. 13(a) represents the tsunami wave entering inside the computational domain. The Fig. 13(b) shows the submersion of the conical island. Figures 13(c) and 13(d) show respectively the large flooding and run-up of the wave and its reflection on the cliff.

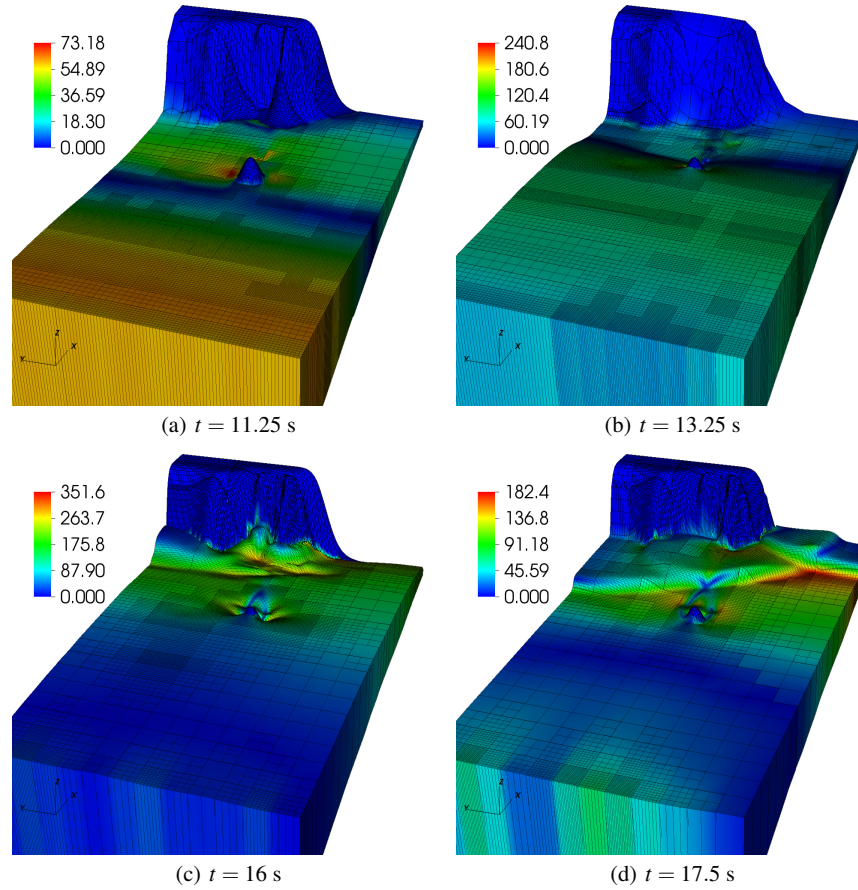


Fig. 13 Numerical water height (coloration is issue from the kinetic energy)

The numerical solutions are now quantitatively confronted to the experimental measures. The obtained comparisons are shown in Fig. 14 at the three waves gauges WG 1 (see Fig. 14(a)), 2 (see Fig. 14(b)) and 3 (see Fig. 14(c)). These figures show that the wave amplitude and the wave shape are accurately predicted at three different location in the computed domain. As emphasized before, dispersive effects being negligible, the obtained results are in a very good agreement. However, it can be noticed in Fig. 14 a small amplitude errors occurring at the beginning of the simulations. This error can be attributed to the lack of accuracy of the prescribed initial condition and boundary condition at $x = 0$. This problem was already encountered by Popinet [23].

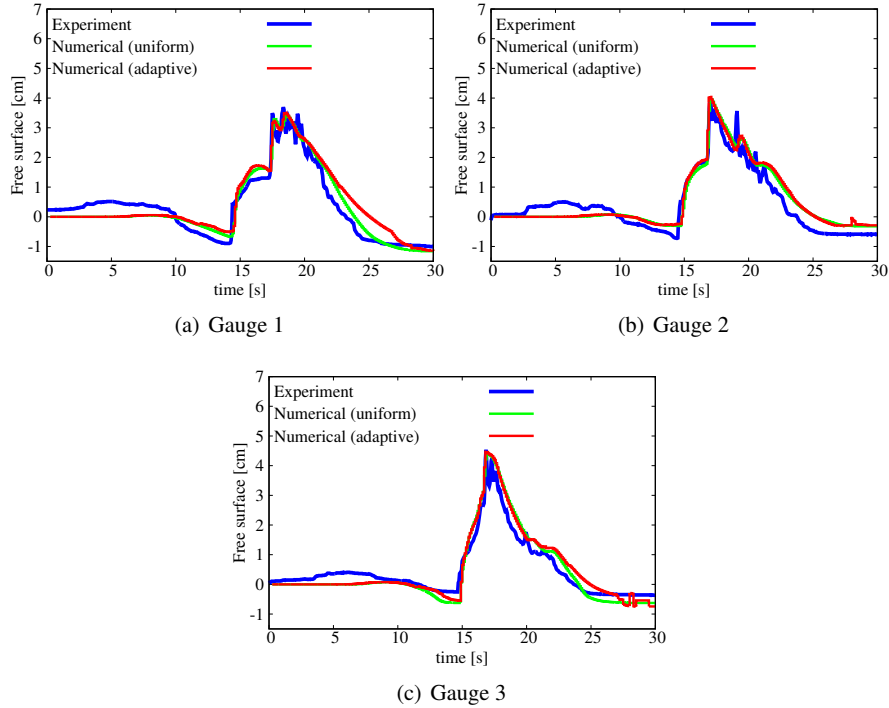


Fig. 14 Free surface results at different positions : experimental data versus numerical simulation with and without mesh adaptivity

4.3.3 AMR performance

To highlight the efficiency of the mesh refinement parameter and the automatic selection of the threshold, the numerical density of entropy production (without absolute value) and the associated mesh are displayed at several times in Fig. 15, (time $t = 11.25$ s in Fig. 15(a), for $t = 13.25$ s in Fig. 15(b), for $t = 16$ s in Fig. 15(c) and for $t = 17.5$ s in Fig. 15(d)). We also display in Fig. 16(a) the evolution of the threshold parameter which selects the region to be refined.

We observe mainly that the wave front and subsequent waves are very well-captured by the criterion and the mesh refinement threshold is well chosen to refine in those areas (see for instance Fig. 13(d) and 15(d)). Between 3 and 7 seconds, the tsunami propagates in the domain involving a grow-up of the number of cells. Then the wave passes through the island and reaches the reef around 15 seconds. After this, the wave is reflected and goes back to the wave maker with a second interaction with the island. The maximum number of cells is reached when the wave is passing on the island and begins to run-up the coast (see Figs. 15(b) and 16(a)), and when the wave goes back to the wave maker. These results are coherent with the

physical process involved since it correspond to the two time where large flow region mixing non negligible multiple flows (smooth and discontinuous) are observed. The overall scales of the tsunami propagation with its reflections are very well captured as displayed in Fig. 15(a), Fig. 15(c) and Fig. 15(d).

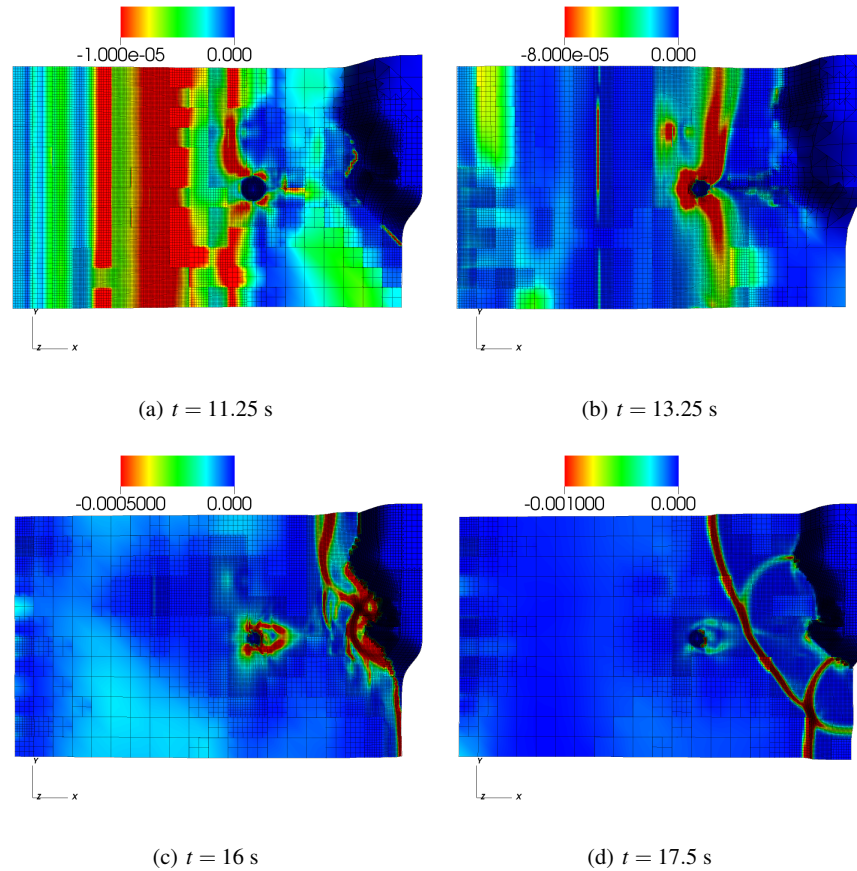


Fig. 15 Adaptive mesh (coloration is issue from the numerical density of entropy production without absolute value)

Finally, the time evolution of the number of cells is represented in Fig. 16(b). Compared to the computation on the uniform grid (see Fig. 14), the adaptive mesh refinement method requires an average of almost 25000 cells against 62000 cells. For the same accuracy, the AMR method allows to save almost 60% of cells with respect to the uniform simulation. Thus, the AMR method allows to speed up the computation almost 3 times.

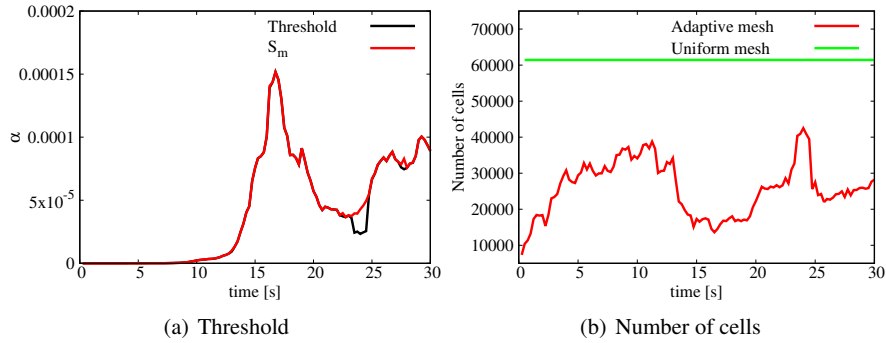


Fig. 16 Time evolution of the mesh refinement threshold and the number of cells

5 Conclusion

In this paper, we have considered the one and two dimensional non linear Saint-Venant system for tsunamis propagation problems, which are well-known to be non-dispersive. Solving accurately this model over very large ranges in spatial scale leads to heavy computational time. Therefore, we have considered an adaptive well-balanced finite volume scheme (based on the hydrostatic reconstruction). To improve the efficiency of the data management a Block-Based strategy is used. The AMR method is also improved using an Automatic Mesh Refinement Threshold based on the decreasing rearrangement of the mesh refinement criterion function [22] yielding to a *parameterless* and robust AMR method.

To test the efficiency of the overall method, and in particular of the automatic threshold technique, we have considered three classical large scale tsunami propagation test cases involving run-up, reflection and/or flooding and drying phenomena. As expected for the non dispersive test cases, we have obtained results in a very good agreement. In the case of weakly dispersive flows, we have obtained some small errors as pointed out before by several authors. However, the presented results can be improved using depth-averaged dispersive model such as the Boussinesq (weakly non linear) or the Green-Naghdi (fully non linear) one for instance.

As emphasized in [22], the use of the automatic threshold allows to localise almost all regions to be refined balancing the computational cost and the accuracy. The Automatic Mesh Refinement Threshold technique makes the AMR method completely *parameterless* and robust while the existing AMR methods are based on a fixed threshold and therefore are test case-dependent. We recall that the automatic threshold is chosen between the smallest local maxima and the global maximum of the mesh refinement criterion.

The overall efficiency and performance of the method has been shown through these three different test cases. Finally, let us highlight that the automatic threshold-

ing method is independent of the numerical method used and the equations solved. Therefore it can be applied for a wide range of numerical applications.

Acknowledgments

This work is partially supported the Project MTM2011-29306-C01-01 from the MICINN (Spain) and the French national research project TANDEM (Tsunamis in the Atlantic and the English channel : definition of the effects through numerical modeling), the French government (Projets Investissement d’Avenir, agreement reference number ANR-11-RSNR-0023-01).

References

1. Altazin, T., Ersoy, M., Golay, F., Sous, D., Yushchenko, L.: Numerical investigation of BB-AMR scheme using entropy production as refinement criterion. *International Journal of Computational Fluid Dynamics* (2016). DOI 10.1080/10618562.2016.1194977. URL <https://hal.archives-ouvertes.fr/hal-01330654>
2. Audusse, E., Bouchut, F., Bristeau, M.O., Klein, R., Perthame, B.: A fast and stable well-balanced scheme with hydrostatic reconstruction for shallow water flows. *SIAM Journal on Scientific Computing* **25**(6), 2050–2065 (2004)
3. Berger, M., Colella, P.: Local adaptive mesh refinement for shock hydrodynamics. *Journal of Computational Physics* **82**(1), 64 – 84 (1989). DOI [http://dx.doi.org/10.1016/0021-9991\(89\)90035-1](http://dx.doi.org/10.1016/0021-9991(89)90035-1). URL <http://www.sciencedirect.com/science/article/pii/S0021999189900351>
4. Berger, M., Oliger, J.: Adaptive mesh refinement for hyperbolic partial differential equations. *J. Comp. Phys.* **53**(3), 484–512 (1984)
5. Bonneton, P.: Analyse physique et modélisation des processus hydrodynamiques en zone de surf. *Oceanis* pp. 157–179 (2003)
6. Coupez, T., Hachem, E.: Solution of high-reynolds incompressible flow with stabilized finite element and adaptive anisotropic meshing. *Computer Methods in Applied Mechanics and Engineering* **267**(0), 65 – 85 (2013). DOI <http://dx.doi.org/10.1016/j.cma.2013.08.004>. URL <http://www.sciencedirect.com/science/article/pii/S0045782513002077>
7. Ersoy, M.: Dimension reduction for incompressible pipe and open channel flow including friction. In: J. Brandts, S. Korotov, M. Krizek, K. Segeth, J. Sistek, T. Vejchodsky (eds.) *Conference Applications of Mathematics 2015, in honor of the birthday anniversaries of Ivo Babuska (90), Milan Prager (85), and Emil Vitasek (85)*. Institute of Mathematics CAS (2015)
8. Ersoy, M., Golay, F., Yushchenko, L.: Adaptive multiscale scheme based on numerical density of entropy production for conservation laws. *Cent. Eur. J. Math.* **11**(8), 1392–1415 (2013). DOI 10.2478/s11533-013-0252-6. URL <http://dx.doi.org/10.2478/s11533-013-0252-6>
9. Eymard, R., T., G., Herbin, R.: Finite volume methods. In: *Handbook of numerical analysis, Vol. VII, Handb. Numer. Anal., VII*, pp. 713–1020. North-Holland, Amsterdam (2000)
10. Fuster, D., Agbaglah, G., Josserand, C., Popinet, S., Zaleski, S.: Numerical simulation of droplets, bubbles and waves: state of the art. *Fluid Dynamics Research* **41**(6), 065,001 (2009)
11. Gerbeau, J.F., Perthame, B.: Derivation of viscous Saint-Venant system for laminar shallow water; numerical validation. *Discrete Cont. Dyn. Syst. Ser. B* **1**(1), 89–102 (2001)

12. Godlewski, E., Raviart, P.: Numerical approximation of hyperbolic systems of conservation laws, *Applied Mathematical Sciences*, vol. 118. Springer-Verlag, New York (1996)
13. Golay, F.: Numerical entropy production and error indicator for compressible flows. *Comptes Rendus Mécanique* **337**(4), 233–237 (2009)
14. Golay, F., Ersoy, M., Yushchenko, L., Sous, D.: Block-based adaptive mesh refinement scheme using numerical density of entropy production for three-dimensional two-fluid flows. *International Journal of Computational Fluid Dynamics* **29**(1), 67–81 (2015)
15. Hachem, E., Feghali, S., Codina, R., Coupez, T.: Immersed stress method for fluid structure interaction using anisotropic mesh adaptation. *International Journal for Numerical Methods in Engineering* **94**(9), 805–825 (2013). DOI 10.1002/nme.4481. URL <http://dx.doi.org/10.1002/nme.4481>
16. Karni, S., Kurganov, A., Petrova, G.: A smoothness indicator for adaptive algorithms for hyperbolic systems. *J. Comp. Phys.* **178**(2), 323–341 (2002)
17. Liu, P.L., Synolakis, C.: Advanced numerical models for simulating tsunami waves and runup, *Advances in Coastal and Ocean Engineering*, vol. 10, liu, philip lf and synolakis, costas edn. School of Civil and Environmental Engineering (2008)
18. Losasso, F., Gibou, F., Fedkiw, R.: Simulating water and smoke with an octree data structure. *ACM Trans. Graph.* **23**(3), 457–462 (2004). DOI 10.1145/1015706.1015745. URL <http://doi.acm.org/10.1145/1015706.1015745>
19. Lynett, P.J., Swigler, D., Son, S., Bryant, D., Socolofsky, S.: Experimental study of solitary wave evolution over a 3d shallow shelf. *Coastal Engineering Proceedings* **1**(32), 1 (2011)
20. Marche, F.: Derivation of a new two-dimensional viscous shallow water model with varying topography, bottom friction and capillary effects. *European Journal of Mechanic. B, Fluids* **26**(1), 49–63 (2007)
21. Min, C., Gibou, F.: A second order accurate level set method on non-graded adaptive cartesian grids. *Journal of Computational Physics* **225**(1), 300 – 321 (2007). DOI <http://dx.doi.org/10.1016/j.jcp.2006.11.034>. URL <http://www.sciencedirect.com/science/article/pii/S0021999106005912>
22. Pons, K., Ersoy, M.: Adaptive mesh refinement method. Part 1: Automatic thresholding based on a distribution function. (2016)
23. Popinet, S.: Quadtree-adaptive tsunami modelling. *Ocean Dynamics* **61**(9), 1261–1285 (2011)
24. Popinet, S.: Adaptive modelling of long-distance wave propagation and fine-scale flooding during the tohoku tsunami. *Nat. Hazards Earth Syst. Sci* **12**(4), 1213–1227 (2012)
25. Roeber, V., Cheung, K.F.: Boussinesq-type model for energetic breaking waves in fringing reef environments. *Coastal Engineering* **70**, 1–20 (2012)
26. Roeber, V., Cheung, K.F., Kobayashi, M.H.: Shock-capturing boussinesq-type model for nearshore wave processes. *Coastal Engineering* **57**(4), 407–423 (2010)
27. Toro, E.F.: Riemann solvers and numerical methods for fluid dynamics, third edn. Springer-Verlag, Berlin (2009). DOI 10.1007/b79761. URL <http://dx.doi.org/10.1007/b79761>
28. Williamschen, M.J., Groth, C.: Parallel anisotropic block-based adaptive mesh refinement algorithm for three-dimensional flows. In: 21st AIAA Computational Fluid Dynamics Conference, June 24-27, San Diego, CA, pp. 1–22 (2013)
29. Yiu, K., Greaves, D., Cruz, S., Saalehi, A., Borthwick, A.: Quadtree grid generation: Information handling, boundary fitting and cfd applications. *Computers & Fluids* **25**(8), 759 – 769 (1996). DOI [http://dx.doi.org/10.1016/S0045-7930\(96\)00029-1](http://dx.doi.org/10.1016/S0045-7930(96)00029-1). URL <http://www.sciencedirect.com/science/article/pii/S0045793096000291>
30. Zhang, M., Wu, W.: A two dimensional hydrodynamic and sediment transport model for dam break based on finite volume method with quadtree grid. *Applied Ocean Research* **33**(4), 297 – 308 (2011). DOI <http://dx.doi.org/10.1016/j.apor.2011.07.004>. URL <http://www.sciencedirect.com/science/article/pii/S0141118711000563>
31. Zheng, Z., Groth, C.: Block-based adaptive mesh refinement finite-volume scheme for hybrid multi-block meshes. In: 7st conference on Computational Fluid Dynamics (ICCFD7), July 9-13, Hawaii, pp. 1–19 (2012)