Journal of Applied Fluid Mechanics, Vol. 16, No. 10, pp. 2054-2068, 2023.
Available online at www.jafmonline.net, ISSN 1735-3572, EISSN 1735-3645.
https://doi.org/10.47176/jafm.16.10.1802

Simulation of Nonlinear Free Surface Waves using a Fixed Grid Method

A. Ebrahimi† and B. Boroomand

Department of Civil Engineering, Isfahan University of Technology, Isfahan 84156-83111, Iran

†*Corresponding Author Email: ali.ebrahimi@cv.iut.ac.ir*

ABSTRACT

The simulation of nonlinear surface waves is of significant importance in safety studies of fluid containers and reservoirs. In this paper, nonlinear free surface flows are simulated using a fixed grid method which employs local exponential basis functions (EBFs). Assuming the flow to be inviscid and irrotational, the velocity potential Laplace's equation is spatially discretized and solved by considering the nonlinear Bernoulli's equation for irrotational flow as the boundary condition on the free surface. The nonlinear boundary conditions are imposed through a semi-implicit iterative time marching. The fixed grid feature of the method, based on a Lagrangian description of fluid flow, allows for retaining the portion of the discretization performed in the first time step for the bulk of the fluid. Thus, the portion which pertains to the regions near the moving boundaries is reprocessed during the time marching. The accuracy and efficiency of the existing solution is shown by simulating various problems such as liquid sloshing induced by external excitation of the reservoir or initial deformed shape of liquid, seiche phenomena and solitary wave propagation in a basin with constant depth or with a step, and comparing the results with those which are analytically available or those from available codes such as Abaqus. The proposed method shows far better stability of the results when compared with those of Abaqus which sometimes exhibit divergence after a relatively large number of time steps. For instance, in the propagation of the considered solitary wave in an infinite-like domain problem, the wave height is calculated by the maximum error of 1.6% and 9% using the present method and Abaqus, respectively.

Article History

Received February 01, 2023
Revised May 29, 2023
Accepted June 17, 2023
Available online July 29, 2023

Keywords:

Moving boundaries
Meshless methods
Exponential basis functions
Potential flow
Velocity potential
Water oscillation
Wave profile

1. INTRODUCTION

Investigating free surface waves of fluids is of importance for designers, since its knowledge helps to increase preparedness and safety of engineering structures. For instance, liquid sloshing in tanker trucks or similar situations can cause overturning moments or other damaging forces. By calculating these moments and forces, it is possible to analyze the stability of the compound system, and if needed, to make changes in its design. Among these changes is the use of baffles in the tanks to change the natural frequency of the system, the liquid's displacement and other flow's characteristics. Consequently, many researchers have investigated such systems under sloshing phenomena. Tuned-liquid dampers (TLDs) are the samples of structural components which require a correct analysis of fluid behavior (see for instance Crowley & Porter, 2012; Love & Tait, 2014; Pandit & Biswal, 2019).

Regarding liquid sloshing in tanks, Abramson (1966) comprehensively investigated the problem using linear theories for the first time. In recent years, the sloshing phenomenon has been revisited using different approaches. Mandal and Maity (2016) analyzed the water sloshing in a rectangular tank using the finite elements method (FEM). In this article, the equation obtained by merging the momentum and continuity equations, and neglecting higher order terms, is solved through a Galerkin approach. The free surface boundary conditions are also considered assuming linear waves during the sloshing. Xue et al. (2019) studied the sloshing in reservoirs with various shapes using the OpenFOAM code. Rawat et al. (2019) investigated the sloshing problem in cylindrical and rectangular tanks using the FEM-based software, Abaqus, and two other numerical approaches. In this research, Bernoulli's equation is used to calculate the pressure while ignoring the higher order terms. Also, the free surface is traced using a volume

NOMENCLATURE

A	amplitude of excitation or wave	$\mathbf{u}$, $\mathbf{u}_s$	velocity vector of fluid/solid walls
$\mathbf{C}$	constant factors matrix	$\mathbf{X}$	global Cartesian coordinate
$\mathbf{E}$	basis functions matrix	X, Y	components of $\mathbf{X}$
$\mathbf{F}$	constants column vector	$\mathbf{x}$	local Cartesian coordinate
f_D, f_N	Dirichlet/Neumann boundary conditions values	x, y	components of $\mathbf{x}$
$\mathbf{g}$	gravity acceleration vector	Δt	time step
$\mathbf{K}$	coefficients matrix	η	free surface elevation
$\mathbf{N}$	shape functions matrix	ρ	fluid density
$\mathbf{n}$	outward normal vector	φ	velocity potential
p	fluid pressure		

fraction tool, which is somewhat similar to the volume of fluid (VOF) method.

Apart from sloshing in tanks, fluid flow simulation plays a prominent role in the studies of natural disasters. The occurrence of an earthquake in the seabed, and the induced vertical displacement, causes the displacement of a huge volume of water, within the order of 1 m of height and 300 km of length, and thus formation and movement of surface waves with the speed of 700 km/hr (Goring, 1978) known as "tsunami". The shape of these waves will change when they enter the shallow water regions near the coastal areas. The produced high-energy waves can cause destructive and deadly natural disasters if they hit industrial and urban areas. However, it is possible to control/reduce the power of the waves by constructing breakwaters if the fluid's behavior is fully understood.

Since 1970, the tsunami phenomenon has been investigated by many researchers using the model of a solitary wave (see Madsen et al., 2008; Domínguez et al., 2019; Chao et al., 2021). For this reason, the propagation of solitary waves in the basins with different bottom's topologies is widely studied. Kânoğlu and Synolakis (1998) examined solitary wave propagation over piecewise linear topography analytically and experimentally. Lo and Young (2004) simulated free surface flows such as solitary wave propagation and collision with rigid walls, interaction of two opposite solitary waves and propagation of a solitary wave passing over a submerged obstacle by using the arbitrary Lagrangian Eulerian (ALE) approach and also by solving velocity-vorticity equations through a weak FEM formulation obtained by the Galerkin method. Nakoulima et al. (2005) presented an analytical solution for the solitary wave passing over periodic topography. Shen and Chan (2008) studied the transformation of the solitary wave on trapezoidal and rectangular submerged obstacles using a numerical method based on the immersed boundary (IB). In this article, the VOF method was used to track the free surface. Also, Hsiao and Lin (2010), Wu et al. (2012), Xiao et al. (2013) and Zarruk et al. (2015) investigated the solitary wave propagation in various basins experimentally and also by using VOF numerical methods. Akbari (2017) studied the solitary wave breaking and its collision with vertical and inclined walls by using an improved version of smoothed particle hydrodynamics (SPH) method. Paprota et al. (2018) used a semi-analytical approach, as well as the SPH method, to investigate the propagation of a solitary wave and its collision with a vertical wall.

On a rather similar line of research, Tripepi et al. (2020) studied the horizontal and vertical forces acting on a submerged square barrier during the propagation of a solitary wave using the SPH method followed by a laboratory simulation for the validation of the results. Magdalena et al. (2020) presented an analytical solution based on the linear shallow wave equations and then used it to examine the reduction of wave's amplitude while the wave is passing over two rectangular obstacles. They compared the results with a numerical solution of the aforementioned equations based on the finite volume method (FVM). Han and Dong (2020), using the SPH method, investigated solitary waves passing over submerged barriers with semi-cylindrical and rectangular shapes. Chao et al. (2021) numerically and experimentally investigated the passage of waves over a single step while considering different dimensions for it. Ghafari et al. (2021) experimentally/numerically compared the wave passage over two submerged barriers. The numerical modeling has been performed using ANSYS Fluent based on the FVM. By using an analytical method and OpenFOAM, Mohapatra et al. (2022) studied the characteristics of a solitary wave passing over a rigid floating structure.

The investigations carried out in the field of fluid dynamics can be classified into three categories of analytical, numerical and laboratory studies or a combination of them. However, analytical solutions are generally limited to specific problems and conditions. Also, laboratory tests require special equipment and thus they are operationally expensive. Therefore, numerical modeling is the most accessible method for investigating engineering problems.

Due to relatively large, or even very large, displacements in most of the aforementioned problems, mesh-based numerical methods suffer from the fundamental defect of mesh distortion. As a result, in recent years, the development and use of mesh-free methods have been the focus of many researchers. Among these methods, as stated earlier, SPH has been used as one of the most powerful and well-known meshless methods in solving many such problems. However, requiring a huge number of particles, considering the limitation due to the Courant number, leads to an enormous computational burden (see Domínguez et al., 2022). It is worth

mentioning that massive parallelization, use of graphics processing units (GPUs) power, and development of codes such as DualSPHysics for providing hardware acceleration possibility are part of the efforts that have been made to compensate for the shortcoming of the SPH method (i.e. increasing the computational power of the available equipment and not necessarily decreasing the computational cost of the method).

Another successful and widely used method to deal with the large displacement of fluid is the VOF, and its derivatives, which are used to track the moving boundaries. This method was presented by Hirt and Nichols (1981) and has been widely used in different researches with various approaches. However, in order to track relatively accurate and sufficiently smooth boundaries, high resolution meshes are needed. Again, this effect in turn leads to high computational cost and storage space.

The objective of this paper is to present a method with a relatively small number of variables (nodal values) for simulating nonlinear waves. In this study, Navier-Stokes equations for fully incompressible inviscid fluid are numerically solved by means of potential theory using the local form of the exponential basis functions (EBFs) method, for spatially discretization of the governing equation, equipped with a second-order time marching scheme to discretize the time domain. The flow is assumed to be irrotational and the free surface dynamic boundary condition is satisfied using the nonlinear Bernoulli's equation for irrotational flows. The solution method is employed to simulate problems such as fluid sloshing due to the external excitation of the container or the initial deformation of the fluid, high amplitude wave movement in a long tank (seiche phenomenon), and the propagation of a solitary wave in a basin with constant depth and with a step. It is worth noting that the EBFs method has been developed in different forms; by Soleimanifar et al. (2014), Shojaei et al. (2015); Mirfatah et al. (2019) and used in the analysis of various engineering problems; by Zandi et al. (2012); Boroomand et al. (2016); Pary Abarghooei and Boroomand (2018); Movahedian et al. (2021); Motamedi et al. (2022). The application of EBFs in construction of absorbing boundary conditions can be found in the works by Shojaei et al. (2019, 2020), Mossaiby et al. (2020), Hermann et al. (2022). In this paper, as a part of the solution procedure to spatially discretize the governing differential equations, we employ the last version of the EBFs method introduced by Mirfatah and Boroomand (2021) which uses a fixed background grid as the location of the main degrees of freedom (DOFs) and satisfies the boundary conditions through a least square procedure. This feature results in the independence of the DOFs' location from the boundaries, and thus, makes it possible to model problems with moving boundaries displaced with relatively large amplitude. As will be shown later, the method is capable of generating accurate geometry in fluid flow problems. The comparison of the obtained results with the available analytical and numerical results shows a good agreement between the two sets despite the use of much less DOFs

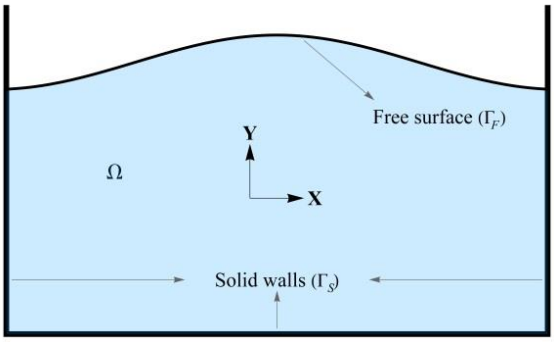

Fig. 1 Domain of a generic problem and its boundaries

and larger time steps compared to the SPH method. Also, the modeling of seiche phenomenon and solitary wave propagation is performed by using the FEM-based Abaqus through which the superior performance of the method, in terms of stability, has been confirmed.

The layout of the paper is as follows: In section 2 the governing equations of the problem are mentioned. Section 3 describes the numerical method used to discretize the governing equations. Section 4 is dedicated to solving various problems of free surface flows and comparing the results, and in section 5, overall conclusions are presented.

2. GOVERNING EQUATIONS

Consider a 2D domain, Ω, occupied by an incompressible inviscid fluid (Fig. 1). Assuming the fluid flow to be irrotational, the well-known potential theory states that the velocity vector, $\mathbf{u}$, has a scalar potential function, φ, such that $\mathbf{u}=\nabla\varphi$ and the potential function satisfies the following equation

$$\nabla^2\varphi=0 \quad \text{in } \Omega \tag{1}$$

in which ∇^2 is the Laplace operator. Also, the boundary conditions on the impermeable solid walls (Γ_S) can be expressed as

$$\mathbf{u}.\mathbf{n}=\mathbf{u}_s.\mathbf{n} \quad \text{or} \quad \partial\varphi/\partial n=\mathbf{u}_s.\mathbf{n} \quad \text{on } \Gamma_S \tag{2}$$

where $\mathbf{n}$ and $\mathbf{u}_s$ are the outward normal vector and velocity vector of the solid wall, respectively. In addition, the nonlinear dynamic boundary condition on the free surface (Γ_F) is written as

$$\mathrm{D}\varphi/\mathrm{D}t=(1/2)\nabla\varphi.\nabla\varphi-gy \quad \text{on } \Gamma_F \tag{3}$$

in which y is the vertical coordinate. Finally, the governing equation and related boundary conditions can be summarized in

$$\begin{cases}\nabla^2\varphi=0 & \text{in } \Omega\\ \partial\varphi/\partial n=\mathbf{u}_s.\mathbf{n} & \text{on } \Gamma_S\\ \varphi=\varphi_F & \text{on } \Gamma_F\\ \mathrm{D}\varphi/\mathrm{D}t=(1/2)\nabla\varphi.\nabla\varphi-gy & \text{on } \Gamma_F\\ \varphi|_{t=0}=\varphi_0 & \text{on } \Gamma_F\end{cases} \tag{4}$$

where, Γ_S and Γ_F are solid boundaries and free surface, respectively, so that $\partial\Omega = \Gamma_S \bigcup \Gamma_F$. Note that φ_0 can be found by the initial velocity, which is zero for flows initiated from rest.

3. NUMERICAL METHOD

In this section, we explain the EBFs method as a meshless approach, to spatially discretize the potential function, and a second-order scheme to discretize the time domain.

3.1 Spatial Discretization, Using the EBFs

Several forms of methods using EBFs have so far been developed by researchers. Using a fixed grid to discretize the domain, and to reduce the sensitivity of the method to the boundaries' placement, is expected to be beneficial for dealing with the problems that have moving boundaries. Accordingly, we employ the version recently presented by Mirfatah and Boroomand (2021) considering its potentials to handle such problems. Here, the method is further equipped with some numerical techniques to adapt to free surface sloshing problems. The solution procedure comprises a spatial solution, at each instance of time, for Eq. (1) along with the following generalized form of the boundary conditions,

$$\begin{cases} \varphi = f_D & \text{on } \Gamma_F \\ \partial\varphi / \partial n = f_N & \text{on } \Gamma_S \end{cases} \tag{5}$$

in which f_D and f_N are used as the simplified forms of the values specified for the Dirichlet and Neumann boundary conditions, respectively, in Eqs. (4). The next section explains the spatial solution procedure.

3.1.1 Spatial Discretization

Consider an arbitrary bounded 2D domain, like the one shown in Fig. 2, whose bounds are illustrated by a set of dots. Here, we name these points as the boundary points and do not define any degree of freedom on them. For spatial discretization and defining DOFs, we scatter a set of main points engulfing the computational domain and its boundaries. A regular grid is employed which in turn contributes to the method's efficiency. In Fig. 2, the main points laying inside and outside the domain are indicated as blue and gray points, respectively. It is worth noting that we define the DOFs only at the main nodes placed inside the domain (internal nodes shown by blue points) and the impact of the external nodes (shown by gray points) will be taken into account during the numerical process through static condensation.

As the next step, to each main point inside the domain we assign a set of neighboring nodes to construct the node's cloud. The neighboring points assigned to a main point may include points both inside and outside the domain. The presented work uses rectangular clouds consisting of eight neighboring nodes for each node, as shown in Fig. 2. The EBFs method in its local form uses the following approximation at each cloud to interpolate the field variable φ

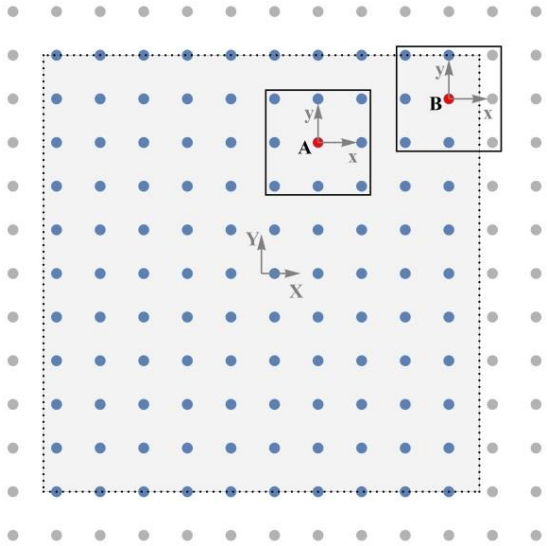

Fig. 2 Spatial discretization

$$\hat{\varphi} = \sum_{i=1}^{m} c_i \, e^{\alpha_i x + \beta_i y} = \mathbf{E}(\mathbf{x})\mathbf{C} \tag{6}$$

$$\mathbf{E}(\mathbf{x}) = \left\{ e^{\alpha_1 x + \beta_1 y}, e^{\alpha_2 x + \beta_2 y}, ..., e^{\alpha_m x + \beta_m y} \right\} \tag{7}$$

$$\mathbf{C} = \left\{ c_1, c_2, ..., c_m \right\}^T \tag{8}$$

where α_i and β_i are constant coefficients, c_i are unknown coefficients and m is the number of basis functions. The main idea of this method is to use some basis functions that satisfy the governing equation while finding the unknown coefficients in such a way that the boundary conditions are satisfied. Thus, the appropriate basis functions for Laplace's Eq. (1) are those which

$$\alpha_i^{\,2} + \beta_i^{\,2} = 0 \tag{9}$$

Evaluation of Eq. (6) at the cloud's nodes (except the central one) leads to

$$\bar{\boldsymbol{\varphi}}_R = \mathbf{E}_R \mathbf{C} \tag{10}$$

where

$$\bar{\boldsymbol{\varphi}}_R = \left\{ \varphi\big|_{\mathbf{x}_1}, \varphi\big|_{\mathbf{x}_2}, \ldots, \varphi\big|_{\mathbf{x}_n} \right\}^{\mathrm{T}} \tag{11}$$

$$\mathbf{E}_R = \left\{ \mathbf{E}(\mathbf{x}_1), \mathbf{E}(\mathbf{x}_2), ..., \mathbf{E}(\mathbf{x}_n) \right\}^T \tag{12}$$

in which $\mathbf{x}_i$ is the local coordinate of the i th node of the cloud with respect to the cloud's central node and n is the number of the cloud's nodes (disregarding the central one). Note that $\mathbf{E}_R$ is an $n \times m$ matrix. Nonetheless, the specific way of choosing the basis functions in the present work leads to $m = n$. Therefore, the unknown coefficients can be evaluated from Eq. (10) as

$$\mathbf{C} = \mathbf{E}_R^{-1} \bar{\boldsymbol{\varphi}}_R \tag{13}$$

Now, Eq. (6) can be rewritten as

$$\hat{\varphi} = \mathbf{N}(\mathbf{x}) \bar{\boldsymbol{\varphi}}_R \tag{14}$$

$$\mathbf{N}(\mathbf{x}) = \mathbf{E}(\mathbf{x})\,\mathbf{E}_R^{-1} \tag{15}$$

in which $\mathbf{N}(\mathbf{x})$ is the shape functions row matrix which, for the sake of conciseness, is presented in the Appendix.

Remark 1. Although the governing differential equations are stated with respect to the global Cartesian coordinate system, $\mathbf{X} = \langle X, Y \rangle$, the solution procedure can be continued with respect to the local Cartesian coordinate system, $\mathbf{x}$, at each cloud. This is true because the only transformation required to map the $\mathbf{X}$ coordinate system to the $\mathbf{x}$ is just a translation and therefore, the Jacobian of this transformation is an identity matrix.■

3.1.2 Construction of the System of Algebraic Equations

As mentioned earlier, the used shape functions satisfy the governing PDE, and therefore, the spatial solution procedure can be completed by applying the boundary conditions and the continuity of the cloud's approximations. The former will be explained in the next section, and the latter can be performed by evaluating Eq. (14) at each cloud in the location of the cloud's center and setting it equal to the nodal value at that location, i.e.

$$\bar{\varphi}_C - \mathbf{N}_C \bar{\boldsymbol{\varphi}}_R = 0 \tag{16}$$

where $\mathbf{N}_C$ and $\bar{\varphi}_C$ are the shape functions vector and nodal value at the cloud's center, respectively. It is worth noting that this procedure is performed for those clouds whose all of their nodes lay inside the domain (for example, the cloud of node A in Fig. 2). Note that in the definition of the shape function, see Eq. (14), the central node was excluded from the set. Therefore, in Eq. (16), the DOF's value associated with the central node is evaluated from the nodes surrounding it.

For the DOFs defined on the central node of the boundary clouds (those that contain at least one external node) Eq. (14) can be divided into two parts as

$$\hat{\varphi} = \mathbf{N}_{int}(\mathbf{x})\ \bar{\boldsymbol{\varphi}}_{int} + \mathbf{N}_{ext}(\mathbf{x})\ \bar{\boldsymbol{\varphi}}_{ext} \tag{17}$$

where the subscripts *int* and *ext* denote the internal and external nodes of the domain, respectively. Since there is no DOF at the external nodes, the term $\bar{\boldsymbol{\varphi}}_{ext}$ should be condensed from the last equation. Hence, this term is evaluated in such a way that the boundary conditions of Eqs. (5) are satisfied by minimizing the following residual

$$\begin{aligned}\Pi = &\sum_{\substack{i \\ (\mathbf{x}_i \in \Gamma_D)}} \left[w_D(\mathbf{x})(\hat{\varphi} - f_D)^2 \right]_{\mathbf{x}=\mathbf{x}_i} \\ &+ \sum_{\substack{i \\ (\mathbf{x}_i \in \Gamma_N)}} \left[w_N(\mathbf{x})(\partial\hat{\varphi}/\partial n - f_N)^2 \right]_{\mathbf{x}=\mathbf{x}_i}\end{aligned} \tag{18}$$

in which Γ_D and Γ_N are the portion of the Dirichlet and Neumann boundaries laying inside the cloud, and $\mathbf{x}_i$ is the local coordinates of the i th boundary point. Also, w_D and w_N are the weight functions related to the Dirichlet and Neumann boundaries, respectively, and are considered as $1/(|\mathbf{x}| + \varepsilon)$ in which ε is a small number to prevent the zero denominators. Substituting Eq. (17) and minimizing the residual as

$$\partial\Pi / \partial\bar{\boldsymbol{\varphi}}_{ext} = \mathbf{0} \tag{19}$$

leads to

$$\mathbf{A}_{int}\ \bar{\boldsymbol{\varphi}}_{int} + \mathbf{A}_{ext}\ \bar{\boldsymbol{\varphi}}_{ext} - \mathbf{T} = \mathbf{0} \tag{20}$$

in which

$$\begin{aligned}\mathbf{T} = &\sum_{\substack{i \\ (\mathbf{x}_i \in \Gamma_D)}} \left[w_D(\mathbf{x})\left(\mathbf{N}_{ext}(\mathbf{x})\right)^T f_D \right]_{\mathbf{x}=\mathbf{x}_i} \\ &+ \sum_{\substack{i \\ (\mathbf{x}_i \in \Gamma_N)}} \left[w_N(\mathbf{x})\left(\partial\mathbf{N}_{ext}(\mathbf{x})/\partial n\right)^T f_N \right]_{\mathbf{x}=\mathbf{x}_i}\end{aligned} \tag{21}$$

$$\begin{aligned}\mathbf{A}_{int} = &\sum_{\substack{i \\ (\mathbf{x}_i \in \Gamma_D)}} \left[w_D(\mathbf{x})\left(\mathbf{N}_{ext}(\mathbf{x})\right)^T \mathbf{N}_{int}(\mathbf{x}) \right]_{\mathbf{x}=\mathbf{x}_i} \\ &+ \sum_{\substack{i \\ (\mathbf{x}_i \in \Gamma_N)}} \left[w_N(\mathbf{x})\left(\frac{\partial\mathbf{N}_{ext}(\mathbf{x})}{\partial n}\right)^T \frac{\partial\mathbf{N}_{int}(\mathbf{x})}{\partial n} \right]_{\mathbf{x}=\mathbf{x}_i}\end{aligned} \tag{22}$$

$$\begin{aligned}\mathbf{A}_{ext} = &\sum_{\substack{i \\ (\mathbf{x}_i \in \Gamma_D)}} \left[w_D(\mathbf{x})\left(\mathbf{N}_{ext}(\mathbf{x})\right)^T \mathbf{N}_{ext}(\mathbf{x}) \right]_{\mathbf{x}=\mathbf{x}_i} \\ &+ \sum_{\substack{i \\ (\mathbf{x}_i \in \Gamma_N)}} \left[w_N(\mathbf{x})\left(\frac{\partial\mathbf{N}_{ext}(\mathbf{x})}{\partial n}\right)^T \frac{\partial\mathbf{N}_{ext}(\mathbf{x})}{\partial n} \right]_{\mathbf{x}=\mathbf{x}_i}\end{aligned} \tag{23}$$

Considering that the $\mathbf{A}_{ext}$ is a square matrix, Eq. (20) can be solved for $\bar{\boldsymbol{\varphi}}_{ext}$ as

$$\bar{\boldsymbol{\varphi}}_{ext} = \left(\mathbf{A}_{ext}\right)^{-1}\left(\mathbf{T} - \mathbf{A}_{int}\ \bar{\boldsymbol{\varphi}}_{int}\right) \tag{24}$$

Substituting Eq. (24) into Eq. (17) leads to

$$\begin{aligned}\hat{\varphi} = &\mathbf{N}_{ext}(\mathbf{x})\mathbf{A}_{ext}^{-1}\mathbf{T} \\ &+\left(\mathbf{N}_{int}(\mathbf{x}) - \mathbf{N}_{ext}(\mathbf{x})\ \mathbf{A}_{ext}^{-1}\mathbf{A}_{int}\right)\bar{\boldsymbol{\varphi}}_{int}\end{aligned} \tag{25}$$

that can be used instead of Eq. (14) for the boundary clouds. Therefore, evaluating the above equation at the cloud's center location generates an equation equivalent to Eq. (16) that corresponds to the DOFs located at the center of the boundary clouds as

$$\bar{\varphi}_c - \left(\mathbf{N}_{int,c} - \mathbf{N}_{ext,c}\mathbf{A}_{ext}^{-1}\mathbf{A}_{int}\right)\bar{\boldsymbol{\varphi}}_{int} = \mathbf{N}_{ext,c}\mathbf{A}_{ext}^{-1}\mathbf{T} \tag{26}$$

where the subscript c denotes the location of the cloud's center. Finally, the whole set of Eqs. (16) and (26) for all clouds appears as a system of algebraic equations as

$$\mathbf{K}\bar{\boldsymbol{\varphi}} = \mathbf{F} \tag{27}$$

where $\mathbf{K}$ is the coefficients matrix, $\bar{\boldsymbol{\varphi}}$ is a column vector consisting of unknown internal nodal values and $\mathbf{F}$ is the constants column vector. Solving Eq. (27) for $\bar{\boldsymbol{\varphi}}$ determines the internal nodal values and the nodal values of the external (but near the bounds) nodes can be evaluated by calculating the average of the values of Eq. (24) found from adjacent clouds containing them. Therefore, through writing Eq. (14) for all clouds, the field variable and its derivatives can be evaluated at any

location in the domain using the weighted average of the solutions within the nearby clouds.

3.2 Time Discretization

The time marching starts from the initial conditions defined in Eqs. (4). Now consider an arbitrary time step, n, between the times t^n, with the boundary points located in the available $\mathbf{x}^n$ geometry and $t^{n+1} = t^n + \Delta t^n$, with the boundary points located in the yet-unknown $\mathbf{x}^{n+1}$ geometry. Through the knowledge of the solid walls' velocity and the values of the potential field on the free surface at the beginning of the time step, $\mathbf{u}_s^n$ and φ_F^n, Eqs. (1) and (5) can be solved to find the velocity field at this time, $\mathbf{u}^n$, as

$$\mathbf{u}^n = \nabla \hat{\varphi}^n \tag{28}$$

Assuming a constant velocity during the time step, the approximate position of the boundary points at the end of the step, which is named as "intermediate geometry", is evaluated as the following equation.

$$\tilde{\mathbf{x}}^{n+1} = \mathbf{x}^n + \mathbf{u}^n \Delta t^n \tag{29}$$

The tilde mark denotes the approximation to the $\mathbf{x}^{n+1}$, and not its exact value. Also, an approximation to the potential values on the free surface at t^{n+1} can be evaluated by using the linear finite difference in the time domain for Eq. (3) as

$$\tilde{\varphi}_{\mathrm{F}}^{n+1} = \varphi_{\mathrm{F}}^n + \left(\frac{1}{2} \nabla \hat{\varphi}.\nabla \hat{\varphi} - gy \right)^n \Delta t^n \tag{30}$$

Now, Eqs. (1) and (5) are solved again at the intermediate geometry, $\tilde{\mathbf{x}}^{n+1}$, using the corresponding boundary conditions, which leads to the attainment of an approximation to the velocity field at the end of the time step in terms of the evaluated potential as

$$\tilde{\mathbf{u}}^{n+1} = \nabla \hat{\tilde{\varphi}}^{n+1} \tag{31}$$

Finally, the position of boundary points at the end of the time step can be evaluated by assuming a linear variation of velocity between $\mathbf{u}^n$ and $\tilde{\mathbf{u}}^{n+1}$ that leads to a second order geometry tracking as

$$\mathbf{x}^{n+1} = \mathbf{x}^n + (1/2)\left(\mathbf{u}^n + \tilde{\mathbf{u}}^{n+1} \right) \Delta t^n \tag{32}$$

Also, the velocity potential values on the free surface can be obtained by a similar approach and using a second-order approximation as

$$\varphi_{\mathrm{F}}^{n+1} = \varphi_{\mathrm{F}}^n + \left(\frac{1}{2} \nabla \hat{\varphi}.\nabla \hat{\varphi} - gy \right)^{n+1/2} \Delta t^n \tag{33}$$

where the superscript $n+1/2$ indicates an average between the quantities found at the beginning of the time step and the intermediate geometry.

Obviously, while marching in time, and as the nonlinearity of the solution becomes prominent, the differences between the values evaluated by two assumptions pertaining to the linear variations and the second-order variations during the time step, Eqs. (30) and (33) also Eqs. (29) and (32), may grow. Hence, this deviation can be used as a criterion for the accuracy of the algorithm. Thus, the following relations must be satisfied

$$\left| \mathbf{x}^{n+1} - \tilde{\mathbf{x}}^{n+1} \right| \le \varepsilon_x \tag{34}$$

$$\left| \varphi_{\mathrm{F}}^{n+1} - \tilde{\varphi}_{\mathrm{F}}^{n+1} \right| \le \varepsilon_\varphi \tag{35}$$

in which ε_x and ε_φ are small values acting as the precision. If the above relations are not satisfied, the final geometry, which is evaluated by a second-order scheme, can now be considered as intermediate geometry as

$$\tilde{\mathbf{x}}^{n+1} = \mathbf{x}^{n+1} \tag{36}$$

$$\tilde{\varphi}_{\mathrm{F}}^{n+1} = \varphi_{\mathrm{F}}^{n+1} \tag{37}$$

and the next iteration is performed by using Eq. (31). It is worth mentioning that the number of iterations at each time step can be used as another criterion to determine the sufficiency of the step length, Δt^n. Therefore, the time step's length will be decreased if the number of iterations exceeds a given value. Moreover, the Courant number defined as

$$c = \left| \mathbf{u} \right|_{max} \ \Delta t / \Delta x_{min} \tag{38}$$

is also used to control the length of the time step. In this regard, the following Courant condition must be satisfied

$$c \le \alpha \tag{39}$$

in which α is a constant. Although $\alpha = 1$ theoretically guarantees the stability of the used numerical scheme, practically, the value of α may be decreased to increase the accuracy of the time discretization scheme as its value is often limited to 0.5 or even 0.1. This value is limited to 0.2 in the present work. Also, we restrict the Courant number to be greater than a minimum value to speed up the simulation.

It is noted that in some cases, particularly problems with rather large deformations, the arrangement of boundary points may become irregular greatly due to the movement of the points in a manner in which some points are separated from each other and some are pressed into together. Hence, another skill which may improve the quality of results is the rearrangement of boundary points. This is performed by equalizing the distances between the nearby boundary points and linearly interpolating the boundary conditions on the new points.

4. NUMERICAL SIMULATIONS

Numerical models related to the free surface fluid flow subjected to various forms of excitation are presented in this section. The fluid, water, is assumed to be inviscid and incompressible and the only body force is gravity. Thus, the fluid motion is due to the presence of gravity and initial velocity or deformation of the fluid or even due to the movement of the fluid container.

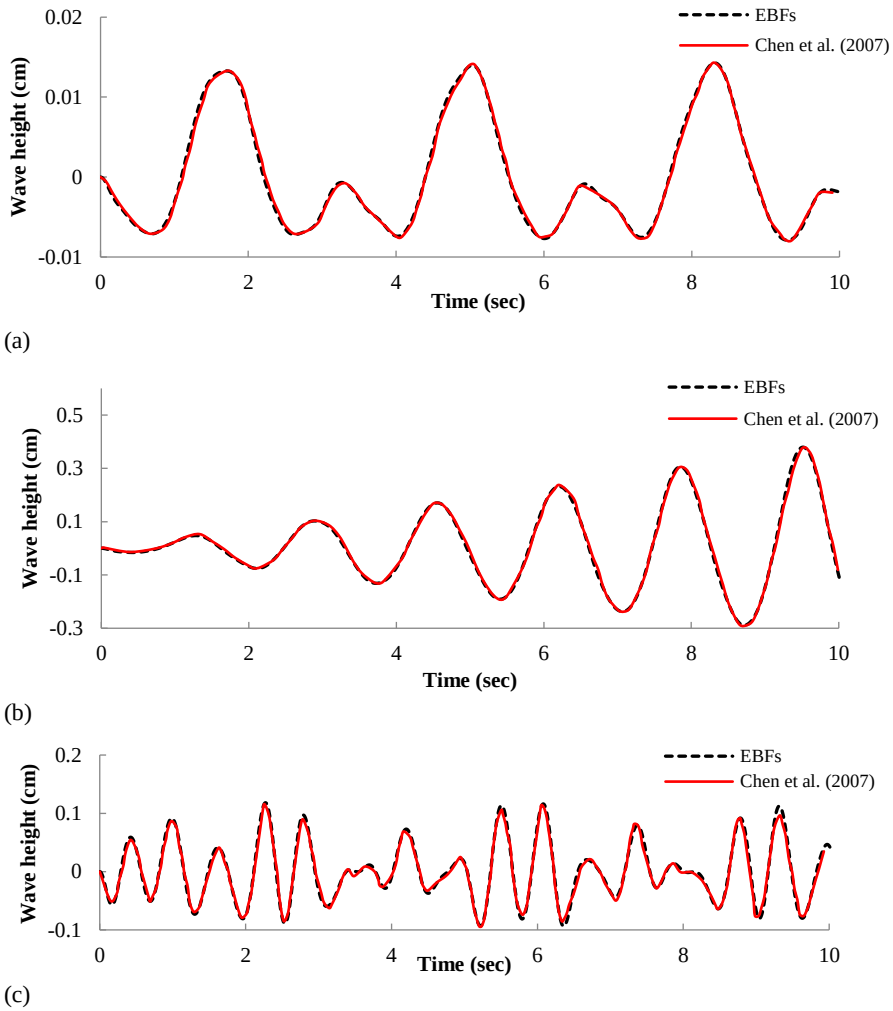

Fig. 3 Wave height at right wall of the rectangular tank subjected to harmonic excitation of frequencies (a) $\omega = 1.9$ **, (b)** $\omega = 3.79$ **and (c)** $\omega = 11.38\ rad/s$

4.1 Rectangular Reservoir Subjected to Harmonic Excitation of Base

This example explains the behavior of a rectangular tank subjected to a harmonic excitation as

$$s(t) = A(1-\cos(\omega t)) \tag{40}$$

The specifications of the model are the same as that given by Chen et al. (2007). i.e., the tank has a length of $L = 80cm$ and a still water depth of $h = 10cm$ while the excitation has an amplitude of $A = 0.04cm$ and various frequencies, ω, of 1.9, 3.79 and 11.38 *rad/s*. These frequencies are chosen to simulate under-resonant, resonant and over-resonant conditions, with the knowledge of the first natural frequency of the container, which is equal to 3.79 *rad/s* for the mentioned tank through the following formula

$$\omega_1 = \sqrt{(\pi g / L)\tanh(\pi h / L)} \tag{41}$$

based on the linear wave theorem.

The example is solved using the presented method by means of a main grid with the point spacing of $\Delta x = \Delta y = 2cm$, boundary points with the spacing of 0.4 times the main grid's node spacing, and time steps of length $\Delta t = 0.01 - 0.02s$ that is 10-20 times the time steps used in the mentioned reference. The wave height at the right wall, obtained by using the present method and that reported in the reference, is shown in Fig. 3.

The results show a good agreement between the results of the present work and those of the mentioned reference despite the fact that larger time steps were used which in turn indicates the efficiency of the proposed time marching.

4.2 Sloshing of Water with Inclined Initial Free Surface

This example deals with the sloshing of water with an initial inclined free surface and zero initial velocity. An inclined free surface can be decomposed into infinite standing waves. Thus, the initial free surface is expressed defined as

$$\eta(x,0) = \sum_{n=1}^{\infty} A_n \sin(k_n x) \tag{42}$$

and the free surface elevation at any time t is evaluated as Eq. (43) using the linear wave theorem

$$\eta(x,t) = \sum_{n=1}^{\infty} A_n \sin(k_n x)\cos(\varpi_n t) \tag{43}$$

where

$$A_n = SL/(n^2\pi^2)(4\sin(n\pi/2) - 2\sin(n\pi)) \tag{44}$$

$$\varpi_n = \sqrt{gk_n \tanh(k_n h)} \tag{45}$$

in which S is the inclined surface slope, L is the tank's length, h is the still water depth and $k_n = n\pi/L$ is the n th mode's wave number. As stated by Lin and Li (2002), it is sufficient to use the first 40 modes to reach an acceptable accuracy. Nonetheless, due to the nonlinear nature of the problem and the linear analytical solution expressed in Eq. (43), the simulation is also performed using the Abaqus software to achieve another nonlinear solution in addition to the present method. For a tank with $L = 1m$, $h = 0.2m$ and $S = 0.02$ the free surface profiles found by using the presented numerical method, the Abaqus model and also the linear analytical solution expressed in Eq. (43) are shown in Fig. 4 named as EBFs, Abaqus and linear wave theory, respectively. The simulation is performed for 7.5 seconds which is about 5 times the first mode's period. The Abaqus model uses 2007 Q4 elements with dimensions of about $1cm$ and the presented method uses a grid of main nodes with the spacing of $2cm$ along X and Y directions, and boundary points with the spacing of about 0.4 times the main grid's node spacing. Also, our code adapts the time steps automatically at the interval of 0.005-0.02 seconds in order to satisfy the conditions stated in Section 3.2. Although a good agreement between the presented results can be seen, the linear solution exhibits some deviations from the nonlinear ones in some cases that can be justified due to the linear nature of this solution which neglects the nonlinear terms. Obviously, larger amplitudes and thus, larger velocities will increase the effect of nonlinear terms and the mentioned deviation. Also, the Abaqus results have an obvious noise in the middle of the free surface profile in the simulation's final frames (Fig. 4h), which in turn indicates the stability of the present method compared to this powerful software.

4.3 Waves with Larger Amplitudes

In this example, chosen from the papers by Zandi et al. (2012) and Lo and Young (2004), the previous problem is analyzed using a trigonometric initial free surface as

$$\eta(x,0) = A\cos(\pi x/L + \pi/2) \tag{46}$$

in a larger reservoir with the length of $L = 98m$ and the still water height of $h = 5m$. For small initial wave's amplitudes, A, the velocity field of fluid can be evaluated by the shallow water theory presented by Neumann and Pierson (1966) as

$$u(x,y,t) = A\sqrt{\frac{g}{h}}\sin\left(\frac{\pi x}{L} + \frac{\pi}{2}\right)\sin\left(\frac{\pi\sqrt{gh}}{L}t\right) \tag{47}$$

$$v(x,y,t) = -\frac{A\sqrt{gh}\pi y}{hL}\cos\left(\frac{\pi x}{L} + \frac{\pi}{2}\right)\sin\left(\frac{\pi\sqrt{gh}}{L}t\right) \tag{48}$$

As such, the wave elevation at the left and right walls can be evaluated using the vertical velocity at $x = \pm L/2$. The simulation is performed using various amplitudes of 0.1, 0.5, 1.0 and 1.5 meters and the results are shown in Fig. 5. The results reported by Zandi et al. (2012) for the amplitudes of 0.1 and 0.5 meters and the results of the Abaqus simulation for all the amplitudes are also presented to provide a better comparison. The aforementioned reference uses 294 boundary points, 400 basis functions and $\Delta t = 0.02s$ while Lo and Young (2004) use 4137 elements and similar time steps. Also, the Abaqus modeling is performed using various element sizes and finally, the value of 0.1 meters (49012 Q4 elements) is chosen as an acceptable optimal value, which provides relatively good accuracy in the case of the smallest amplitude, $A = 0.1m$. For larger amplitudes, the value of 0.5 meters is chosen as the optimal element size. The presented method uses a grid of main nodes with the spacing of $\Delta x = \Delta y = 1m$, boundary points with the spacing of 0.2 times the main nodes spacing and large time steps of $\Delta t = 0.02 - 0.64s$ which is possible thanks to the adaptive time marching and the low nonlinearity of the problem in the small amplitudes (the initial value set to $\Delta t = 0.02s$, the code increased it to $0.64s$ and finally, used the fixed value of $\Delta t = 0.32s$ from $t = 3.4s$ onwards). The larger amplitudes have been simulated using smaller time steps at the interval of $\Delta t = 0.01 - 0.32s$.

As can be observed, as the excitation amplitude increases from 0.1 to 0.5 meters, the difference between the analytical and numerical results increases, which can be justified due to the increase in the nonlinear effects of the problem. Therefore, while the results of shallow water theory become invalid for the larger amplitudes, the simulation is performed using the Abaqus software to provide another measure to verify the accuracy of the obtained results by the presented method. Very good agreement between the results of Abaqus and those of the present method is observable in Fig. 5. This is true except for the cases of the small amplitude of 0.1 and the rather large amplitude of 1.5 meters. In the former case, the Abaqus results become noisy at some intervals. This effect probably occurs because of the nature of the FEM method that requires small elements to accurately capture the small deformations during the time. Whereas, this greatly increases the computational cost due to the large dimensions of the reservoir. Also, as the time marching proceeds, the results of Abaqus in the mentioned case deviate from the other solutions. Nevertheless, the good agreement between the results of the presented method and the those reported by Zandi et al. (2012) indicates the right path of the solution.

In the case of the rather large amplitude of 1.5 meters, the results meet a little difference at some extremes. But, refining the Abaqus mesh, to check the accuracy and convergence of its results, leads to numerical instability resulting from mesh distortion which is predictable due to the rather large displacements of the mesh.

4.4 Solitary Waves in an Infinite-Like Domain

The movement of fluid particles subjected to initial velocity and gravity in a very long tank is investigated in this section. The free surface elevation and initial horizontal velocity are considered similar to those mentioned in the reference Li et al. (2012) as

$$\eta(x,t) = A\,\mathrm{sech}^2\left(\sqrt{3A/\left(4d^3\right)}\left(x - x_0 - ct\right)\right) \tag{49}$$

$$u = \sqrt{g/d}\,\eta(x,0) \tag{50}$$

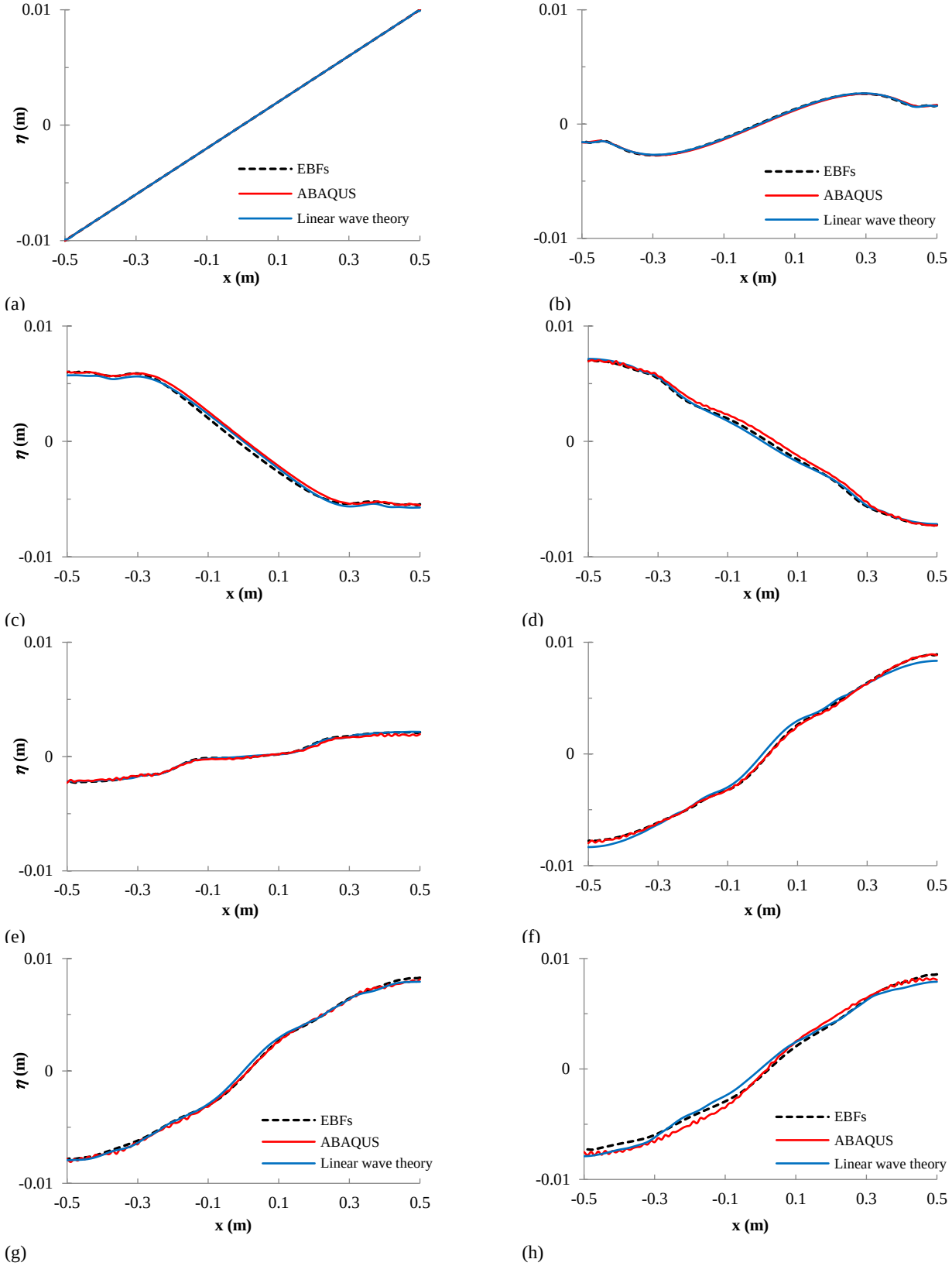

Fig. 4 Free surface profile of the tank with initial inclined surface at the time (a) $t = 0$ **, (b)** $t = 0.3$ **, (c)** $t = 0.6$ **, (d)** $t = 0.9$ **, (e)** $t = 1.2$ **, (f)** $t = 1.5$ **, (g)** $t = 4.5$ **and (h)** $t = 7.5$ (sec)

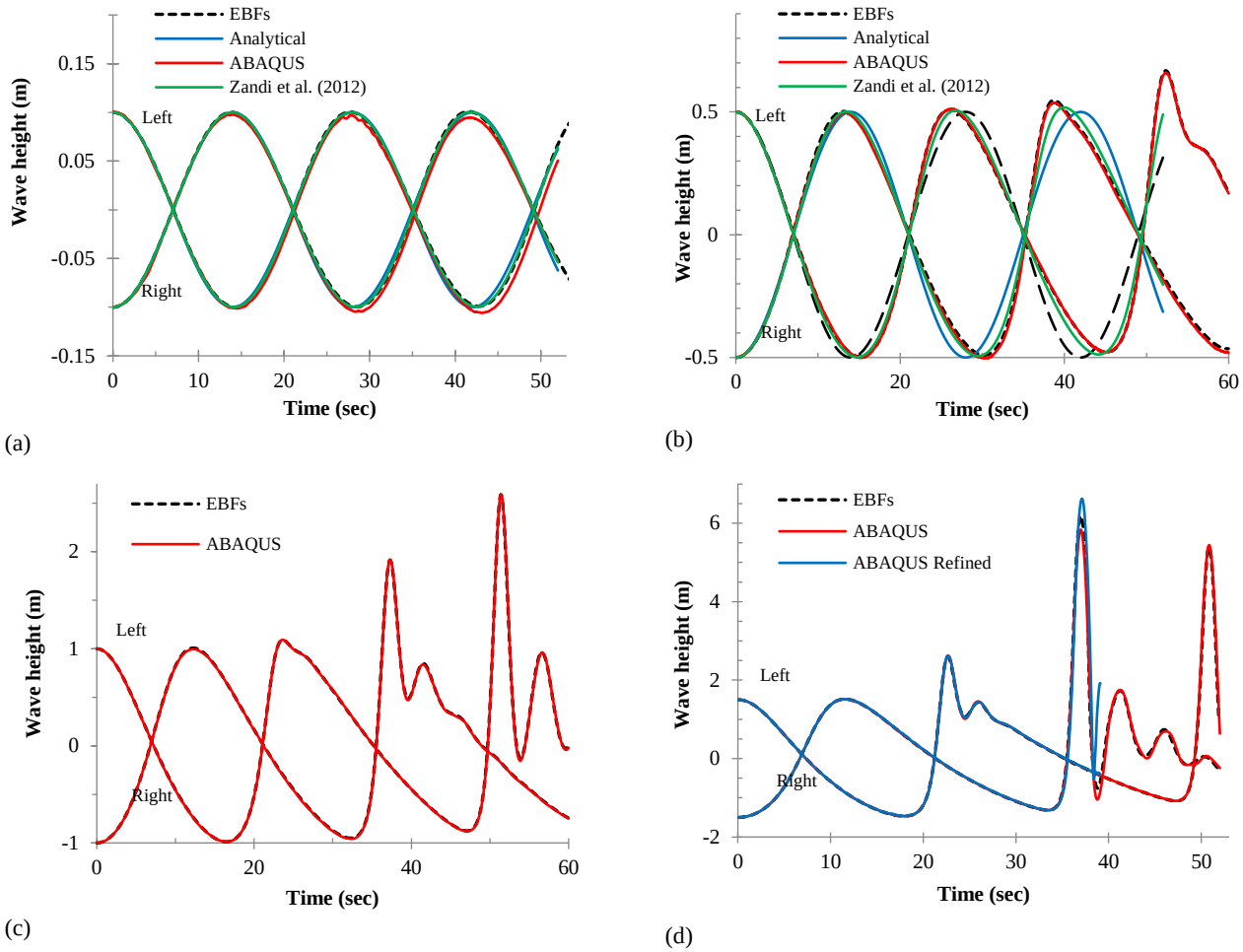

Fig. 5 Wave height on the left and right walls of a tank with initial trigonometric surface of amplitude (a) 0.1, (b) 0.5, (c) 1.0 and (d) 1.5*m*

where x_0 is the initial position of the solitary wave's crest, d is the water depth in the absence of wave, $c=\sqrt{g(d+A)}$ is the solitary wave celerity and the other parameters are the same as those in the previous examples. Considering the condition of lengthy-reservoir is due to the validation of the non-reflection boundary condition that states there is no reflected wave from walls within the domain. By such assumptions, the solitary wave should travel steadily while maintaining its initial shape. Therefore, monitoring the solitary wave shape, as time goes on, will provide a suitable tool for judging on the accuracy and stability of the solution method.

In the present example, a numerical wave tank with the water depth of $d=0.2m$, length of $L=150d$ and $A=0.1825d=0.0365m$ is simulated using the Abaqus software (15032 Q4 elements with an approximate size of 0.02 meters). Also, the simulation is done using the present method (a grid of nodes with spacing of $\Delta y=0.05m$ in order to place 4 main nodes in the height of the tank, $\Delta x=0.1m$, $\Delta t=0.01s$, 1505 initial DOFs and boundary points of spacing 0.1 times the minimum of main nodes distances) and took around 7.5 hours on a 64 bit computer (CPU specification: Intel Core i7-4790k). Figure 6 shows the free surface profiles obtained by the present method, Abaqus and the analytical solution (Eq. 49). In this figure, the coordinate's origin lies on the middle of the tank's length and depth, the initial position of solitary wave's crest is $x_0=-9m$ that is 24 meters far away from the right wall mimicking the non-reflecting walls. It is noted that this problem was solved using a WCSPH model in a 24 meters long tank and approximately 250,000 particles, which took around 6 days, as reported in the reference Li et al. (2012).

As can be seen, proceeding the time will decrease the quality of Abaqus results; while, the present method maintains the shape of the solitary wave with a good agreement up to the end of the simulation. To establish a quantitative comparison, the crest's height at different times is reported in Table 1, which indicates the good accuracy of the presented method's results while approximating the exact value of 0.0365*m*.

Table 1 Comparison of wave amplitude at different times using Abaqus and present method

Time (sec)	EBFs		Abaqus	
	Wave height (m)	Error (%)	Wave height (m)	Error (%)
2.5	0.0366	0.27	0.0361	1.1
5.0	0.0363	0.55	0.0353	3.29
7.5	0.0361	1.1	0.0346	5.21
10	0.0359	1.64	0.0332	9.04

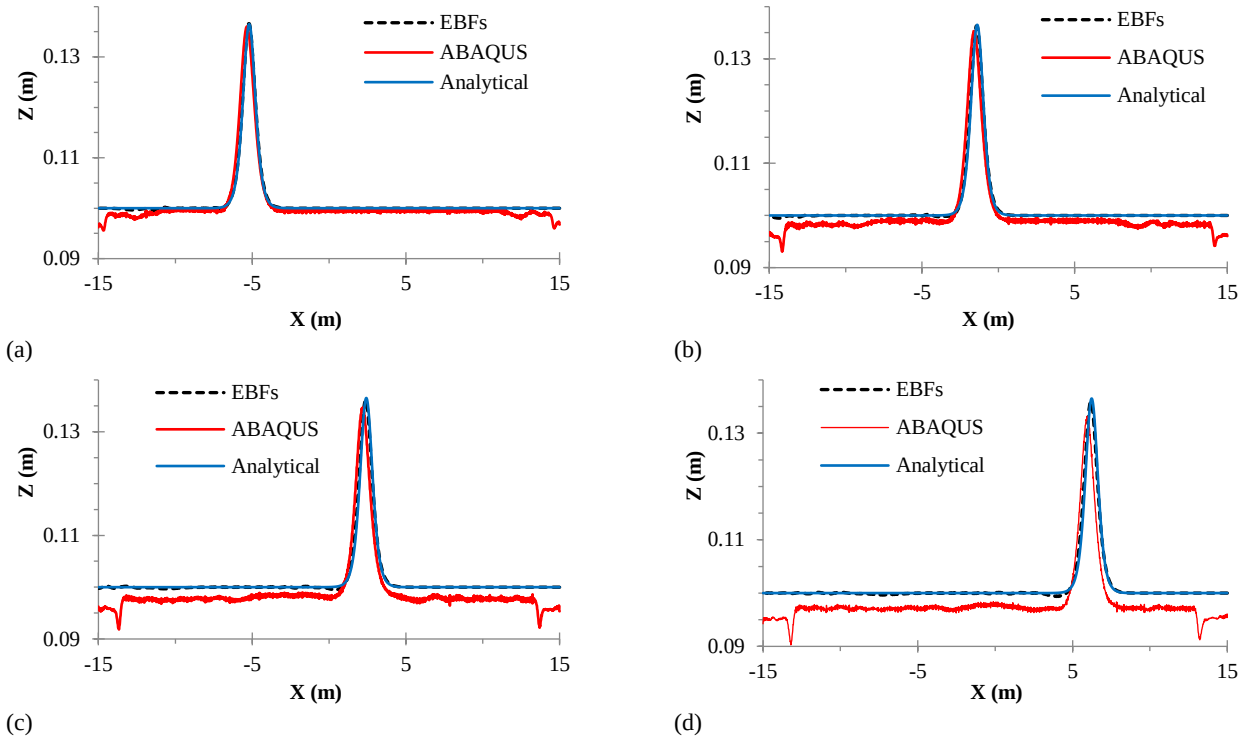

Fig. 6 Free surface profile of solitary wave propagation in an infinite-like domain at the times (a)2.5, (b) 5.0, (c) 7.5 and (d) 10 seconds

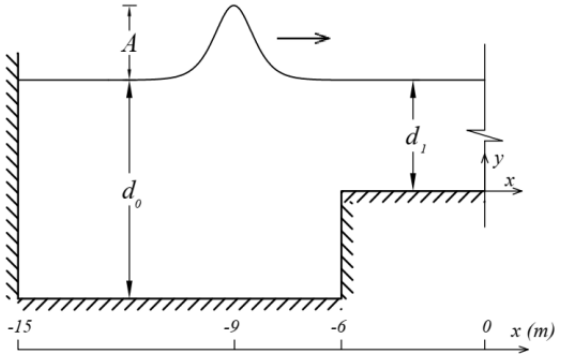

Fig. 7 Initial geometry of solitary wave passing over a step

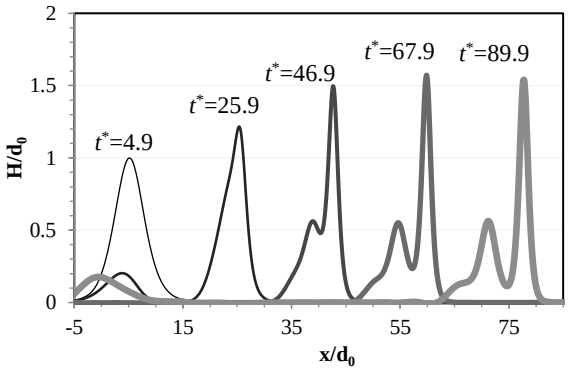

Fig. 8 Formation of solitons after passing a non-breaking solitary wave over a step

4.5 Solitary Wave Passing Over a Step, Fission Phenomenon

The abrupt changes in the water depth like the one illustrated in Fig. 7 influences the solitary wave passing over the step. This leads to the fission phenomenon, which means decomposition of the initial wave into some subsequent waves (called "solitons"). The abrupt change in the water depth and the specific conditions at the step location lead to critical conditions for the mesh-based methods such as the FEM used in the Abaqus which result in the mesh distortion and termination of the simulation. Hence, we could not simulate such a problem with Abaqus due to its limitations.

In this example, which is simulated in the reference Li et al. (2012) using an SPH approach, the propagated wave in a tank with abrupt change in the water depth, as shown in Fig. 7 with $d_1 = 0.5d_0$, is tracked. The wave's specifications are the same as those expressed in Eqs. (49) and (50) in which $A = 0.104d_0$ and $d = d_0 = 0.2m$. It is noted that d_0 and d_1 are the depth of the water before and after the step position, respectively.

The solution of this example is performed using the present method by means of a main grid with node spacing of $\Delta x = \Delta y = 0.05m$, boundary points with spacing of 0.1 times the main point spacing and adaptive time steps as $\Delta t = 0.005 - 0.02s$. The problem has also been solved by Li et al. (2012) using a set of particles with initial spacing of $0.005m$. Figure 8 shows the solitons in an area around the step at the normalized times of 0.7, 3.7, 6.7, 9.7 and 12.8 of the simulation (the normalized time corresponds to the dimensionless parameter $t^* = t\sqrt{g/d_0}$). The evolution of wave is also shown in Fig. 9 which provides a comprehensive overview of the state of wave movement at different times.

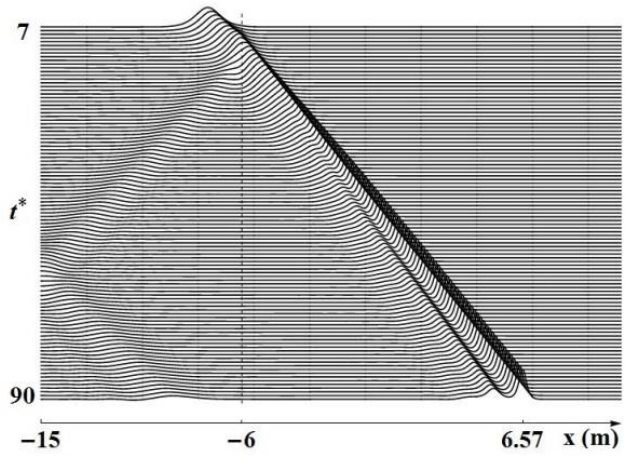

Fig. 9 Evolution of solitary wave after passing over the step (dashed line)

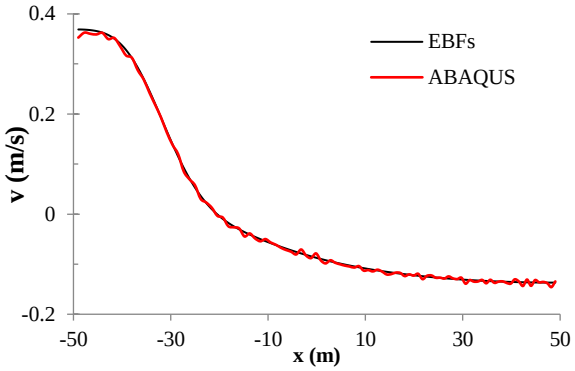

Fig. 10 Vertical velocity of free surface at t=20*s* using Abaqus and present method

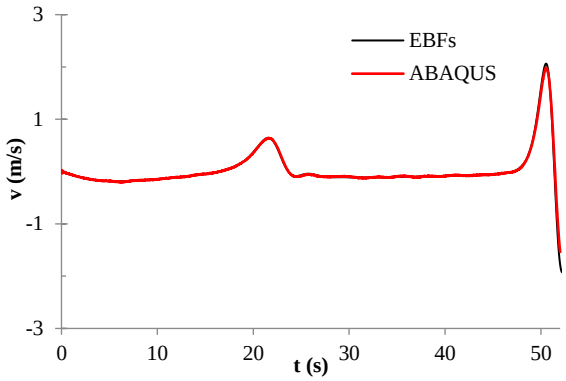

Fig. 11 Vertical velocity of the upper left corner of free surface

As can be observed, the formation of solitons and the movement of the secondary waves are clearly discernible. Also, after passing the wave over the step, a part of the wave is separated from the main wave, travels in the opposite direction and reflects from the left wall after hitting it. This effect is not observed in a mesh-based simulation, as the mentioned reference states. This example clearly demonstrates the capabilities of the presented method compared with some well-established ones.

5. CONCLUSIONS

In this paper, free surface flows of incompressible inviscid fluids have been studied numerically through a fixed grid method. Among the salient features of the proposed method is that it is truly meshfree, and also, it has relatively low sensitivity to the arrangement of the boundary points. The latter feature not only increases the method's versatility in dealing with boundaries with various configurations, but also increases the results' stability in the solution of problems with moving boundaries. In contrast to mesh-based methods, the fact that the discretization does not need degrees of freedom for the boundary points further improves the versatility of the method in the solution of problems with very larger deformations.

Another feature of the presented method is that it provides a relatively smooth and noise-free solution by using exponential functions along with some techniques to increase the solution's smoothness. Among these techniques are the use of interpolation functions for calculating the approximate values and also an averaging process for finding such values from the adjacent clouds. Also, despite the zero-order continuity of the approximated field, it can be seen that the obtained solution has smooth derivatives. For example, Fig. 10 demonstrates the vertical velocity of the free surface in a problem with a rather large wave amplitude, i.e. a wave with an amplitude of $A=1.0m$ at $t=20s$. For such a problem good agreement is observed between the obtained and available results (see Fig. 5-c). In the same path, Fig. 11 shows the variation of the vertical velocity of the upper left point of the free surface versus the time. It is observable that the graphs related to Abaqus generally have significant amount of noise in time and space which adversely affects the accuracy and stability of the solution. Fig. 11 also confirms the effectiveness of the spatial and time discretization used in the proposed method. As far as the time discretization scheme is concerned, it has been observed that the corrective adaptive time marching algorithm can make the solution stable to an acceptable extent. An important point in this regard is the solution's iteration which apparently increases the computational cost. But, regarding the automatic selection of the length of the time steps, it was finally observed that this increase in the cost is compensated by the increase in the length of the time steps. The reader may note that the semi-implicit nature of the algorithm allows the user to employ larger time steps. In addition, checking the difference between the first two consecutive intermediate geometries, which are obtained based on two assumptions of constant and linear velocity during the time step, can play the role of a criterion to find the number of required time steps. Obviously, in problems with lower amounts of nonlinearity, or at some instances of time in more general nonlinear problems in which the whole system exhibits nearly linear behavior, the two assumptions lead to some close results and thus there will be no need of numerous iterations. Therefore, the algorithm's ability to check this criterion can decrease the computational cost. Finally, a combination of checking processes (i.e. for the Courant

number, the number of required iterations and the difference between two consecutively-evaluated intermediate geometries in a time step) makes a sort of balance between the accuracy and the computational cost.

In summary, due to the use of the exponential bases which satisfy the governing equation, and regarding the aforementioned points pertaining to the smoothness and stability of the results, the proposed method can be considered effective compared to the methods available in the literature. Nevertheless, reduction of the computational cost, e.g. in the form of using parallel processing, or further development of the method, e.g. for dealing with multiphase media, can be considered in the future research works. Moreover, the fixed grid feature of the method may make it an ideal candidate for dealing with more general problems involving moving boundaries.

CONFLICT OF INTEREST

The authors declare that they have no competing interest.

AUTHORS CONTRIBUTION

A. Ebrahimi: Writing - original draft, methodology, validation, software; **B. Boroomand**: Supervision, review & editing, Methodology.

REFERENCES

Abramson, H. N. (1966). *The dynamic behavior of liquids in moving containers, with applications to space vehicle technology*. NASA SP-106, Washington.

Akbari, H. (2017). Simulation of wave overtopping using an improved SPH method. *Coastal Engineering*, 126, 51–68. https://doi.org/10.1016/j.coastaleng.2017.04.010

Boroomand, B., Bazazzadeh, S., & Zandi, S. M. (2016). On the use of Laplace's equation for pressure and a mesh-free method for 3D simulation of nonlinear sloshing in tanks. *Ocean Engineering*, 122, 54–67. https://doi.org/10.1016/j.oceaneng.2016.06.019

Chao, W. T., Liang, S. J., Young, C. C., & Ting, C. L. (2021). Interactions of solitary wave with a submerged step: experiments and simulations. *Water*, 13(9), 1302. https://doi.org/10.3390/w13091302

Chen, Y. H., Hwang, W. S., & Ko, C. H. (2007). Sloshing behaviours of rectangular and cylindrical liquid tanks subjected to harmonic and seismic excitations. *Earthquake Engineering & Structural Dynamics*, 36(12), 1701–1717. https://doi.org/10.1002/eqe.713

Crowley, S., & Porter, R. (2012). An analysis of screen arrangements for a tuned liquid damper. *Journal of Fluids and Structures*, 34, 291–309. https://doi.org/10.1016/j.jfluidstructs.2012.06.001

Domínguez, J. M., Altomare, C., Gonzalez-Cao, J., & Lomonaco, P. (2019). Towards a more complete tool for coastal engineering: solitary wave generation, propagation and breaking in an SPH-based model. *Coastal Engineering Journal*, 61(1), 15–40. https://doi.org/10.1080/21664250.2018.1560682

Domínguez, J. M., Fourtakes, G., Altomare, C., Canelas, R. B., Tafuni, A., García-Feal, O., Martínez-Estévez, I., Mokos, A., Vacondio, R., Crespo, A. J. C., Rogers, B. D., Stansby, P. K., & Gómez-Gesteira, M. (2022). State-of-the-art SPH solver DualSPHysics: from fluid dynamics to multiphysics problems. *Computational Particle Mechanics*, 9(5), 867–895. https://doi.org/10.1007/s40571-021-00404-2

Ghafari, A., Tavakoli, M. R., Nili-Ahmadabadi, M., Teimouri, K., & Kim, K. C. (2021). Investigation of interaction between solitary wave and two submerged rectangular obstacles. *Ocean Engineering*, 237, 109659. https://doi.org/10.1016/j.oceaneng.2021.109659

Goring, D. G. (1978) *Tsunamis - the propagation of long waves onto a shelf*. Pasadena, CA: California Institute of Technology.

Han, X., & Dong, S. (2020). Interaction of solitary wave with submerged breakwater by smoothed particle hydrodynamics. *Ocean Engineering*, 216, 108108. https://doi.org/10.1016/j.oceaneng.2020.108108

Hermann, A., Shojaei, A., Steglich, D., Höche, D., Zeller-Plumhoff, B., & Cyron, C. J. (2022). Combining peridynamic and finite element simulations to capture the corrosion of degradable bone implants and to predict their residual strength. *International Journal of Mechanical Sciences*, 220, 107143. https://doi.org/10.1016/j.ijmecsci.2022.107143

Hirt, C. W., & Nichols, B. D. (1981). Volume of fluid (VOF) method for the dynamics of free boundaries. *Journal of Computational Physics*, 39(1), 201–225. https://doi.org/10.1016/0021-9991(81)90145-5

Hsiao, S. C., & Lin, T. C. (2010). Tsunami-like solitary waves impinging and overtopping an impermeable seawall: Experiment and RANS modeling. *Coastal Engineering*, 57(1), 1–18. https://doi.org/10.1016/j.coastaleng.2009.08.004

Kânoğlu, U., & Synolakis, C. E. (1998). Long wave runup on piecewise linear topographies. *Journal of Fluid Mechanics*, 374, 1–28. https://doi.org/10.1017/S0022112098002468

Li, J., Liu, H., Gong, K., Tan, S. K., & Shao, S. (2012). SPH modeling of solitary wave fissions over uneven bottoms. *Coastal Engineering*, 60, 261–275. https://doi.org/10.1016/j.coastaleng.2011.10.006

Lin, P., & Li, C. W. (2002). A σ-coordinate three-dimensional numerical model for surface wave propagation. *International Journal for Numerical Methods in Fluids*, 38(11), 1045–1068. https://doi.org/10.1002/fld.258

Lo, D. C., & Young, D. L. (2004). Arbitrary Lagrangian–Eulerian finite element analysis of free surface flow using a velocity–vorticity formulation. *Journal of Computational Physics*, 195(1), 175–201. https://doi.org/10.1016/j.jcp.2003.09.019

Love, J. S., & Tait, M. J., (2014). Linearized sloshing model for 2D tuned liquid dampers with modified bottom geometries. *Canadian Journal of Civil Engineering*, 41(2), 106–117. https://doi.org/10.1139/cjce-2013-0106

Madsen, P. A., Fuhrman, D. R., & Schäffer, H. A. (2008). On the solitary wave paradigm for tsunamis. *Journal of Geophysical Research: Oceans*, 113(C12). https://doi.org/10.1029/2008JC004932

Magdalena, I., Atras, M. F., Sembiring, L., Nugroho, M. A., Labay, R. S. B., & Roque, M. P. (2020). Wave Transmission by Rectangular Submerged Breakwaters. *Computation*, 8, 56. https://doi.org/10.3390/computation8020056

Mandal, K. K., & Maity, D. (2016). Nonlinear finite element analysis of water in rectangular tank. *Ocean Engineering*, 121, 592–601. https://doi.org/10.1016/j.oceaneng.2016.05.048

Mirfatah, S. M., & Boroomand, B. (2021). On the simulation of image-based cellular materials in a meshless style. *Computers & Mathematics with Applications*, 100, 99–125. https://doi.org/10.1016/j.camwa.2021.08.021

Mirfatah, S. M., Boroomand, B., & Soleimanifar, E. (2019). On the solution of 3D problems in physics: From the geometry definition in CAD to the solution by a meshless method. *Journal of Computational Physics*, 393, 351–374. https://doi.org/10.1016/j.jcp.2019.05.007

Mohapatra, S. C., Islam, H., Hallak, T. S., & Soares, C. G. (2022). Solitary wave interaction with a floating pontoon bbased on boussinesq model and CFD-based simulations. *Journal of Marine Science and Engineering*, 10, 1251. https://doi.org/10.3390/jmse10091251

Mossaiby, F., Shojaei, A., Boroomand, B., Zaccariotto, M., & Galvanetto, U. (2020). Local Dirichlet-type absorbing boundary conditions for transient elastic wave propagation problems. *Computer Methods in Applied Mechanics and Engineering*, 362, 112856. https://doi.org/10.1016/j.cma.2020.112856

Motamedi, A. R., Boroomand, B., & Noormohammadi, N. (2022). A Trefftz based meshfree local method for bending analysis of arbitrarily shaped laminated composite and isotropic plates. *Engineering Analysis with Boundary Elements*, 143, 237–262. https://doi.org/10.1016/j.enganabound.2022.05.018

Movahedian, B., Zohravi, P., Mansouri, S., & Boroomand, B. (2021). Wave propagation in two dimensional structures: An efficient solution method in time domain using exponential basis functions. *Computers & Structures*, 243, 106375. https://doi.org/10.1016/j.compstruc.2020.106375

Nakoulima, O., Zahibo, N., Pelinovsky, E., Talipova, T., & Kurkin, A. (2005). Solitary wave dynamics in shallow water over periodic topography. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 15(3), 037107. https://doi.org/10.1063/1.1984492

Neumann, G., & Pierson, W. J. (1966) *Principles of physical oceanography*. Prentice Hall, New York.

Pandit, A. R., & Biswal, K. C. (2019). Seismic control of structures using sloped bottom tuned liquid damper. *International Journal of Structural Stability and Dynamics*, 19(09), 1950096. https://doi.org/10.1142/S0219455419500962

Paprota, M., Staroszczyk, R., & Sulisz, W. (2018). Eulerian and Lagrangian modelling of a solitary wave attack on a seawall. *Journal of Hydro-environment Research*, 19, 189–197. https://doi.org/10.1016/j.jher.2017.09.001

Pary Abarghooei, H., & Boroomand, B. (2018). Simulating fluid and structure interaction using exponential basis functions. *Journal of Applied Fluid Mechanics*, 11(3), 787–799. https://doi.org/10.29252/jafm.11.03.28347

Rawat, A., Mittal, V., Chakraborty, T., & Matsagar, V. (2019). Earthquake induced sloshing and hydrodynamic pressures in rigid liquid storage tanks analyzed by coupled acoustic-structural and Euler-Lagrange methods. *Thin-Walled Structures*, 134, 333–346. https://doi.org/10.1016/j.tws.2018.10.016

Shen, L., & Chan, E. S. (2008). Numerical simulation of fluid–structure interaction using a combined volume of fluid and immersed boundary method. *Ocean Engineering*, 35(8–9), 939–952. https://doi.org/10.1016/j.oceaneng.2008.01.013

Shojaei, A., Boroomand, B., & Mossaiby, F. (2015). A simple meshless method for challenging engineering problems. *Engineering Computations*, 32(6), 1567–1600. https://doi.org/10.1108/EC-06-2014-0131

Shojaei, A., Hermann, A., Seleson, P., & Cyron, C. J. (2020). Dirichlet absorbing boundary conditions for classical and peridynamic diffusion-type models. *Computational Mechanics*, 66(4), 773–793. https://doi.org/10.1007/s00466-020-01879-1

Shojaei, A., Mossaiby, F., Zaccariotto, M., & Galvanetto, U. (2019). A local collocation method to construct Dirichlet-type absorbing boundary conditions for transient scalar wave propagation problems. *Computer Methods in Applied Mechanics and Engineering*, 356, 629–651. https://doi.org/10.1016/j.cma.2019.07.033

Soleimanifar, E., Boroomand, B., & Mossaiby, F. (2014). A meshless method using local exponential basis functions with weak continuity up to a desired order. *Computational Mechanics*, 53(6), 1355–1374. https://doi.org/10.1007/s00466-014-0979-3

Tripepi, G., Aristodemo, F., Meringolo, D. D., Gurnari, L., & Filianoti, P. (2020). Hydrodynamic forces induced by a solitary wave interacting with a submerged square barrier: Physical tests and δ-LES-SPH simulations. *Coastal Engineering*, 158, 103690. https://doi.org/10.1016/j.coastaleng.2020.103690

Wu, Y. T., Hsiao, S. C., Huang, Z. C., & Hwang, K. S. (2012). Propagation of solitary waves over a bottom-

mounted barrier. *Coastal Engineering*, 62, 31–47. https://doi.org/10.1016/j.coastaleng.2012.01.002

Xiao, H., Huang, W., Tao, J., & Liu, C. (2013). Numerical modeling of wave–current forces acting on horizontal cylinder of marine structures by VOF method. *Ocean Engineering*, 67, 58–67. https://doi.org/10.1016/j.oceaneng.2013.01.027

Xue, M. A., Chen, Y., Zheng, J., Qian, L., & Yoan, X. (2019). Fluid dynamics analysis of sloshing pressure distribution in storage vessels of different shapes. *Ocean Engineering*, 192, 106582. https://doi.org/10.1016/j.oceaneng.2019.106582

Zandi, S. M., Boroomand, B., & Soghrati, S. (2012). Exponential basis functions in solution of incompressible fluid problems with moving free surfaces. *Journal of Computational Physics*, 231(2), 505–527. https://doi.org/10.1016/j.jcp.2012.06.036

Zarruk, G. A., Cowen, E. A., Wu, T. R., & Liu, P. F. (2015). Vortex shedding and evolution induced by a solitary wave propagating over a submerged cylindrical structure. *Journal of Fluids and Structures*, 52, 181–198. https://doi.org/10.1016/j.jfluidstructs.2014.11.001

APPENDIX

Consider a typical cloud consisting of the central node and eight adjacent nodes. Now, we need eight basis functions corresponding to eight adjacent nodes with respect to the local coordinate system, $\mathbf{x}=\{x,y\}$ and so, for each adjacent node we define an ancillary coordinate system, $\mathbf{x}'=\{x',y'\}$ as seen in Fig. A-1.

Then the rational exponential basis function can be written as

$$E(x',y')=G(x').H(y') \tag{A-1}$$

where $G(x')$ is an exponential function as

$$G(x')=e^{\alpha_n x'} \tag{A-2}$$

and $H(y')$ is the Fourier expansion of a hat function, like $T(y')$, as

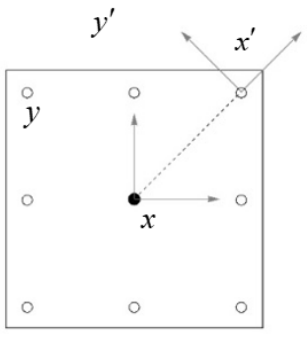

Fig. A-1 A typical cloud and its local coordinates.

$$T(y')=\begin{cases}1+y'/(\kappa_1\times l_B) & -\kappa_1\times l_B\le y'\le 0\\ 1-y'/(\kappa_2\times l_C) & 0\le y'\le \kappa_2\times l_C\end{cases} \tag{A-3}$$

In which l_B and l_C are the minimum non-zero distance between the considered adjacent node and the other auxiliary nodes along $-y'$ and y' directions, respectively. Also, the parameters κ_1 and κ_2 determine the shape of the hat function. Therefore, the function $H(y')$ can be written as

$$H(y')=\sum_{j=-\infty}^{\infty}c_j e^{i\beta_j y'} \tag{A-4}$$

and thus

$$E(x',y')=\sum_{j=-\infty}^{\infty}c_j e^{\alpha_j x'}e^{i\beta_j y'} \tag{A-5}$$

$$c_j=\begin{cases}\dfrac{l}{2(j\pi)^2}\left(\dfrac{1-e^{i\beta_j\kappa_1 l_B}}{\kappa_1 l_B}+\dfrac{1-e^{-i\beta_j\kappa_2 l_C}}{\kappa_2 l_C}\right), & j\neq 0\\ (\kappa_1 l_B+\kappa_2 l_C)/4l, & j=0\end{cases} \tag{A-6}$$

where $\beta_j=j\pi/l$, $l=\kappa\times\max\{l_B,l_C\}$ and κ is a parameter that determines the length of the zero part of the Fourier expansion and satisfies the sampling theorem. Also, the coefficient α_j can be evaluated from Eq. (9) as

$$\alpha_j=|\beta_j| \tag{A-7}$$

in order to have an ascending trend along x' direction. Now, we can construct the shape functions using Eq. (15) and mapping the $\mathbf{x}'$ to the $\mathbf{x}$ coordinate system. However, we can add $\mathbf{p}=\{1,x,y\}$ to consider the double roots of Eq. (9) and improve the shape functions behavior. This can be done by using the following approximation instead of Eq. (6)

$$\hat{\varphi}=c_1+c_2x+c_3y+\sum_{i=1}^{n}E_i(\mathbf{x})d_i=\mathbf{p}^T\mathbf{c}+\mathbf{E}(\mathbf{x})\mathbf{d} \tag{A-8}$$

in which c_i and d_i are the new coefficients that differ from the previously used ones. The procedure initiates with finding the c_i coefficients so that the approximation passes near the nodal values using a least squares scheme and passes the exact values with the aid of the d_i coefficients. This procedure finally leads to an equation similar to Eq. (14) in which

$$\begin{aligned}\mathbf{N}(\mathbf{x})=&\,\mathbf{p}(\mathbf{x})\left(\mathbf{P}^T\mathbf{P}\right)^{-1}\mathbf{P}^T\\ &+\mathbf{E}(\mathbf{x})(\mathbf{E}_R)^{-1}\left(\mathbf{I}-\mathbf{P}\left(\mathbf{P}^T\mathbf{P}\right)^{-1}\mathbf{P}^T\right)\end{aligned} \tag{A-9}$$

where

$$\mathbf{P}=\left\langle \mathbf{p}|_{\mathbf{x}_1}\quad \mathbf{p}|_{\mathbf{x}_2}\quad \dots\quad \mathbf{p}|_{\mathbf{x}_n}\right\rangle^T \tag{A-10}$$

The reader may refer to Mirfatah and Boroomand (2021) for more detailed explanations.