INTERNATIONAL JOURNAL FOR NUMERICAL METHODS IN FLUIDS
Int. J. Numer. Meth. Fluids 2010; **62**:1013–1046
Published online 28 April 2009 in Wiley InterScience (www.interscience.wiley.com). DOI: 10.1002/fld.2056

Multi-scale simulation with a hybrid Boussinesq-RANS hydrodynamic model

K. I. Sitanggang and P. J. Lynett*, †

Coastal and Ocean Engineering Division, Zachry Department of Civil Engineering, 3136 Texas A&M University, College Station, TX 77843-3136, U.S.A.

SUMMARY

A hybrid wave model is developed for simulation of water wave propagation from deep water to shoreline. The constituent wave models are the irrotational, 1-D horizontal Boussinesq and 2-D vertical Reynolds-averaged Navier–Stokes (RANS). The models are two-way coupled, and the interface is placed at a location where turbulence is relatively small. Boundary conditions on the interfacing side of each model are provided by its counterpart model through data exchange. Prior to the exchange, a data transformation step is carried out due to the differences in physical variables and approximations employed in both models. The hybrid model is tested for both accuracy and speedup performance. Tests consisting of idealized solitary and standing wave motions and wave overtopping of nearshore structures show that: (1) the simulation results of the current hybrid model compare well with the idealized data, experimental data, and pure RANS model results and (2) the hybrid model saves computational time by a factor proportional to the reduction in the size of the RANS model domain. Finally, a large-scale tsunami simulation is provided for a numerical setup that is practically unapproachable using RANS model alone; not only does the hybrid model offer more rapid simulation of relatively small-scale problems, it provides an opportunity to examine very large total domains with the fine resolution typical of RANS simulations.

Received 2 August 2008; Revised 24 February 2009; Accepted 26 February 2009

KEY WORDS: Boussinesq model; breaking wave; coupled wave model; hybrid model; RANS model; solitary wave

1. INTRODUCTION

Most of the currently available ocean wave models are developed based on a single set of governing equations applied on the entire computational domain. While model tests have shown agreement

*Correspondence to: P. J. Lynett, Coastal and Ocean Engineering Division, Zachry Department of Civil Engineering, 3136 Texas A&M University, College Station, TX 77843-3136, U.S.A.
†E-mail: plynett@civil.tamu.edu

with various target scenarios, the applicability of such models for more general purpose applications such as the simulation of offshore-to-shoreline wave propagation, which includes both the nonbreaking and breaking wave processes, is still limited due to the physical assumptions in the models. For instance, depth-integrated equations using potential flow assumptions are one of the more commonly employed equations/assumptions in existing wave models. The implementation of such models on a certain domain is valid only if the flow regime is far from the high-intensity turbulence area such as the nearshore breaking zone. Although the application of such models with the help of the ad hoc turbulence [1, 2] may be pushed further nearshore to include the wave-breaking processes, caution should be taken for the implementation of a more general and complex 3-D nearshore bathymetry. Another set of equations that are currently widely used for wave modeling is the turbulence-closed Reynolds-averaged Navier–Stokes (RANS) equations. The models developed based on these equations are well suited for breaking wave and wave–structure simulations [3, 4]. The implementation of this model, however, is usually confined to the nearshore zone where a relatively large number of grid points and fine mesh are needed to accurately capture the turbulence. This incurs expensive computational effort. Hence, extending the implementation of the model to a larger domain is often not practically feasible.

This paper describes the simultaneous use of two wave models that belong to the first and second aforementioned wave models for wave simulation in the ocean. The two models are the Boussinesq-equation type and the RANS-equation-based wave models. In this study, the scope is limited to coupling the 1-D Boussinesq and 2-D RANS wave models. To allow for application on a wider range of wave nonlinearity, the fully nonlinear Boussinesq-equation model [5, 6] is employed. For the second model, we use a RANS-based model with a two-equation turbulence closure scheme [3]. The two models are two-way coupled, and so act as if they are a single model working on a continuous domain. In the coupling implementation, the Boussinesq model is applied in the non-breaking zone and the RANS model in the breaking/high-turbulence zone. The two models share a common domain interface for exchanging data, used as boundary conditions in the models. By coupling the two models, accurate large-scale wave simulation using a coarse grid and 'simple' physics in the deep-to-intermediate water and fine grid and detailed physics in the nearshore area is computationally feasible. In summary, the coupled hybrid model bridges the two widely used wave models.

The rest of the paper is organized as follows. First, we briefly explain the two constituent wave models along with their numerical solution procedures. Second, the coupling method and the data exchange for boundary conditions are explained in detail. Third, we test the hybrid model against several scenarios for validation and relative speedup. Finally, a large-scale tsunami simulation is provided for a numerical setup that is practically unapproachable using RANS model alone; demonstrating the potential of the hybrid model to efficiently tackle multi-scale problems.

2. BOUSSINESQ-EQUATION WAVE MODEL

2.1. Governing equation

Consider wave propagation from offshore to the shoreline as depicted in Figure 1. Depending on the wave condition, this sea area is unequally divided into nonbreaking (offshore) and breaking (nearshore) zones. Turbulence is assumed to be small inside the nonbreaking zone and is therefore neglected. Using this assumption along with the incompressibility of water, the flow inside the

DOI: 10.1002/fld

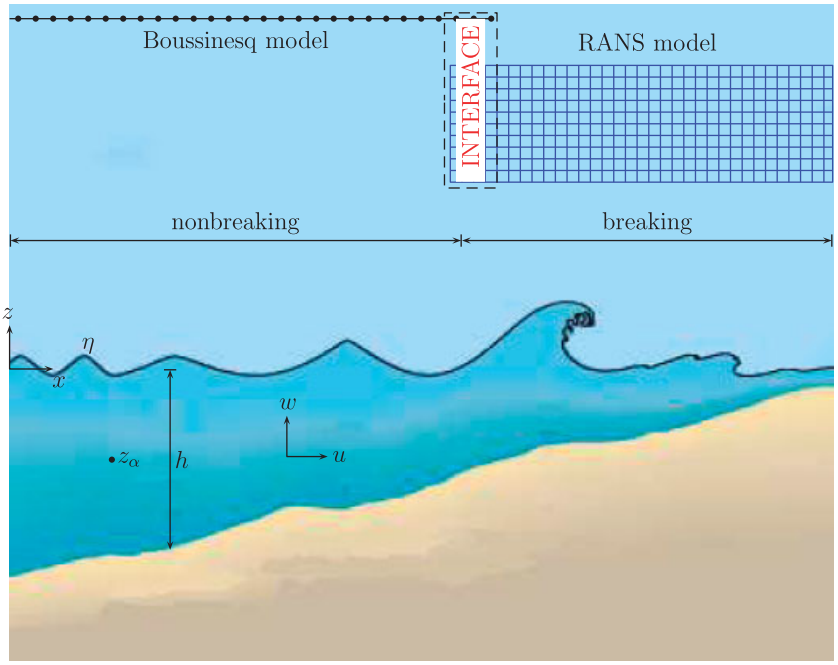

Figure 1. Wave propagation from offshore to shoreline (adapted from [7]). In the hybrid model, the 1-D Boussinesq equation is employed to model the 2-D nonbreaking wave motion and the 2-D RANS equation to model the 2-D wave-breaking motion. The boundary condition in each model is provided through data exchange on the interface area.

nonbreaking zone can be modeled using the continuity and Euler equations:

$$\mu^2 u_x + w_z = 0 \tag{1}$$

$$\mu^2 u_t + \varepsilon\mu^2 u u_x + \varepsilon w u_z + \mu^2 p_x = 0 \tag{2}$$

$$\varepsilon w_t + \varepsilon^2 u w_x + \frac{\varepsilon^2}{\mu^2} w w_z + \varepsilon p_z + 1 = 0 \tag{3}$$

The above equations are dimensionless where the coordinate system, time, velocity, and pressure are nondimensionalized as

$$x = \frac{x'}{l}, \quad z = \frac{z'}{h_0}, \quad t = \frac{\sqrt{gh_0}}{l} t', \quad u = \frac{h_0}{a_0\sqrt{gh_0}} u', \quad w = \frac{h_0^2}{a_0 l\sqrt{gh_0}} w', \quad p = \frac{p'}{\rho g a_0}$$

All variables with subscript are the dimensional forms. The variable g is gravity, h_0 is the characteristic depth, l and a_0 are wave length and typical wave amplitude. Related with the last three variables are the nonlinearity and frequency dispersion parameters, defined as $\varepsilon = a_0/h_0$ and $\mu = h_0/l$, respectively.

For a single-valued free surface problem, (1)–(3) can be vertically integrated with respect to z over the entire depth, $\int_{-h(x)}^{\eta(x,t)} (\ldots)\,\mathrm{d}z$, to reduce the model from 2-D to a one horizontal dimension

DOI: 10.1002/fld

(1HD) problem. This requires the imposition of the dynamic, $p=0$, and kinematic, $w=\eta_t+u\eta_x$, free surface boundary conditions at $z=\eta$, and zero normal velocity, $w=-uh_x$, at the bottom, $z=-h$. From here, certain assumptions are made to derive the Boussinesq equations. First, it is assumed that the frequency dispersion is small, $O(\mu^2)\ll 1$. The wave nonlinearity, however, is not assumed to be small. Second, to perform the integration the horizontal velocity is expressed in terms of the velocity at some reference level, z_α, as used in [8], via a Taylor series expansion

$$u(x,z,t)=u(x,z_\alpha,t)+(z-z_\alpha)u_z(x,z_\alpha,t)+\frac{(z-z_\alpha)^2}{2}u_{zz}(x,z_\alpha,t)+\cdots \tag{4}$$

Note that while the assumptions made here permit the creation of an extremely efficient wave model, they govern the limits of application of the model, and thus play an important role in the development of the coupled system.

Retaining higher-order terms in (4), as done in [6], and performing the integration (with some algebraic manipulations) gives, in dimensional form, the 1HD continuity

$$\frac{\partial H}{\partial t}+\frac{\partial (Hu_\alpha)}{\partial x}-\frac{\partial}{\partial x}\left\{H\left[\left(\frac{1}{6}(\eta^2-\eta h+h^2)-\frac{1}{2}z_\alpha^2\right)\frac{\partial S}{\partial x}+\left(\frac{1}{2}(\eta-h)-z_\alpha\right)\frac{\partial T}{\partial x}\right]\right\}=0 \tag{5}$$

and momentum equation

$$\begin{aligned}&\frac{\partial u_\alpha}{\partial t}+\frac{1}{2}\frac{\partial u_\alpha^2}{\partial x}+g\frac{\partial \eta}{\partial x}+\frac{\partial}{\partial t}\left\{\frac{1}{2}z_\alpha^2\frac{\partial S}{\partial x}+z_\alpha\frac{\partial T}{\partial x}-\frac{\partial}{\partial x}\left(\frac{1}{2}\eta^2 S+\eta T\right)\right\}\\&+\frac{\partial}{\partial x}\left\{\frac{\partial \eta}{\partial t}(T+\eta S)+(z_\alpha-\eta)\left(u_\alpha\frac{\partial T}{\partial x}\right)+\frac{1}{2}(z_\alpha^2-\eta^2)\left(u_\alpha\frac{\partial S}{\partial x}\right)\right.\\&\left.+\frac{1}{2}(T+\eta S)^2\right\}=0\end{aligned} \tag{6}$$

where u_α is the velocity at the reference level, $H=h+\eta$, $S=\partial u_\alpha/\partial x$, and $T=\partial(hu_\alpha)/\partial x+\partial h/\partial t$.

The use of z_α in the integration has resulted in a frequency dispersion term, the third term in (5), which would not be produced if we had to use the depth-averaged velocity as the velocity variable [9]. This variable is set to $z_\alpha=-0.531h$, where h is the depth, to obtain the best fit between the linear dispersion of the model and the exact dispersion for a wide range of water depths [8].

The linear-dispersion accuracy limit of the above equation is near $kh=3.0$, where k is the wave number and the nonlinearity is accurate up to $kh=1.0$ [5].

As will be explained later, the Boussinesq model passes the vertical profiles of the velocities, $u(z)$ and $w(z)$, to the RANS model in the data exchange between the two models. Although the 2-D problem has been converted to the 1HD, this profiles are obtainable from the original 2-D equations. The former can be obtained from the z-integration of the irrotationality condition, $w_x-u_z=0$:

$$u=u_\alpha-\tfrac{1}{2}(z^2-z_\alpha^2)u_{\alpha xx}-(z-z_\alpha)(hu_\alpha)_{xx} \tag{7}$$

DOI: 10.1002/fld

while the latter is found from (1):

$$w = -zu_{\alpha x} - (hu_\alpha)_x \tag{8}$$

In both derivations, higher-order term, $O(\mu^4)$, is truncated.

2.2. Numerical solution

Equations (5) and (6) are coupled, first order in time differential equations for the free surface elevation, η, and reference velocity, u_α. A few methods are available to integrate the equations in time, such as the Runge–Kutta, the Richardson extrapolation, and the predictor–corrector methods. Owing to the complexity of the equations, we choose the last method for solving the equations. One variant of this method is the higher-order Adams–Bashforth–Moulton predictor–corrector [10], which has good stability properties. For comparison, we also have implemented a Runge–Kutta method, which demonstrates instability for most of the cases tested. We conjecture that the Adams–Bashforth–Moulton predictor–corrector is the better choice for solving (5) and (6), and is also the approach most commonly found in the literature for solving these equations (e.g. [6]).

Following terms arrangement used in [5] for better numerical stability recasts (5) and (6) as

$$\eta_t = E(u_\alpha, \eta, h) \tag{9}$$

$$U_t = F(u_\alpha, \eta, h) \tag{10}$$

where

$$\begin{aligned} E = &-h_t - [(\eta+h)u_\alpha]_x \\ &+\{(\eta+h)[(\tfrac{1}{6}(\eta^2-\eta h+h^2)-\tfrac{1}{2}z_\alpha^2)S_x+(\tfrac{1}{2}(\eta-h)-z_\alpha)T_x]\}_x \end{aligned} \tag{11}$$

$$\begin{aligned} F = &-\tfrac{1}{2}(u_\alpha^2)_x - g\eta_x - z_\alpha h_{xtt} - z_{\alpha t}h_{xt} + (\eta h_{tt})_x - [E(\eta S+T)]_x \\ &-[\tfrac{1}{2}(z_\alpha^2-\eta^2)u_\alpha S_x]_x - [(z_\alpha-\eta)u_\alpha T_x]_x - \tfrac{1}{2}[(T+\eta S)^2]_x \end{aligned} \tag{12}$$

$$U = u_\alpha + \tfrac{1}{2}(z_\alpha^2-\eta^2)u_{\alpha xx} + (z_\alpha-\eta)(hu_\alpha)_{xx} - \eta_x[\eta u_{\alpha x} + (hu_\alpha)_x] \tag{13}$$

In the predictor step, the explicit third-order Adams–Bashforth method is used to solve (5) and (6):

$$\eta_i^{n+1} = \eta_i^n + \tfrac{1}{12}\Delta t(23E_i^n - 16E_i^{n-1} + 5E_i^{n-2}) \tag{14}$$

$$U_i^{n+1} = U_i^n + \tfrac{1}{12}\Delta t(23F_i^n - 16F_i^{n-1} + 5F_i^{n-2}) \tag{15}$$

where superscripts indicate the time level and subscripts represent the grid or spatial location. The implicit fourth-order Adams–Moulton method is used in the corrector step:

$$\eta_i^{n+1} = \eta_i^n + \tfrac{1}{24}\Delta t(9E_i^{n+1} + 19E_i^n - 5E_i^{n-1} + E_i^{n-2}) \tag{16}$$

$$U_i^{n+1} = U_i^n + \tfrac{1}{24}\Delta t(9F_i^{n+1} + 19F_i^n - 5F_i^{n-1} + F_i^{n-2}) \tag{17}$$

DOI: 10.1002/fld

Algorithm 1. Algorithm to numerically solve the Boussinesq equations.

1. Calculate $\left\{\eta_i^{n+1} | i=1, \ldots, N\right\}$ from (14).
2. Calculate $\left\{U_i^{n+1} | i=1, \ldots, N\right\}$ from (15).
3. Solve the tridiagonal system of equations (13) for $\left\{u_{\alpha i}^{n+1} \middle| i=1, \ldots, N\right\}$.
4. With the new η_i^{n+1} and $u_{\alpha i}^{n+1}$, calculate $\left\{E_i^{n+1} | i=1, \ldots, N\right\}$ from (11).
5. Calculate $\left\{\eta_i^{n+1} | i=1, \ldots, N\right\}$ from (16).
6. With the new η_i^{n+1} and $u_{\alpha i}^{n+1}$, calculate $\left\{F_i^{n+1} | i=1, \ldots, N\right\}$ from (12).
7. Calculate $\left\{U_i^{n+1} | i=1, \ldots, N\right\}$ from (17).
8. Solve the tridiagonal system of equations (13) for $\left\{u_{\alpha i}^{n+1} \middle| i=1, \ldots, N\right\}$.
9. Calculate

$$r_\eta = \frac{\sum_{i=1}^{N} |(\eta_i^{n+1})_{\text{old}} - (\eta_i^{n+1})_{\text{new}}|}{\sum_{i=1}^{N} |(\eta_i^{n+1})_{\text{old}}|}$$

and

$$r_u = \frac{\sum_{i=1}^{N} |(u_{\alpha i}^{n+1})_{\text{old}} - (u_{\alpha i}^{n+1})_{\text{new}}|}{\sum_{i=1}^{N} |(u_{\alpha i}^{n+1})_{\text{old}}|}$$

where subscripts old and new denote the corresponding previous and current values of the iterative process 4–14.

10. **if** $r_\eta < 10^{-4}$ and $r_u < 10^{-4}$ **then**
11. Advance to the next time level and repeat procedure until end time is reached.
12. **else**
13. Goto step: 4.
14. **end if**

The first-order spatial derivatives in (11)–(13) are calculated using fourth-order accurate spatial finite difference method, while higher derivatives are differenced to second-order accurate. With this scheme, the calculation of (14)–(17) requires η and u_α from five points to the left and right of point-i, creating an 11-point stencil. Note that both η_i and $u_{\alpha i}$ are calculated at the same location.

Let there be N points in the domain, numbered $-4, \ldots, N+5$ (Figure 2), with the extra five points on the left and right sides of the domain acting as 'imaginary' points used to enforce the boundary conditions. Suppose that the free surface elevation and velocity, $\{\eta_i^m, u_{\alpha i}^m | i=-4, \ldots, N+5, m=n, n-1, n-2\}$, are given and hence $\{E_i^m, F_i^m | i=1, \ldots, N, m=n, n-1, n-2\}$ can be calculated. Given these initial values, the Boussinesq equations are solved using the procedure presented in Algorithm 1.

The corrector steps, i.e. steps 4 to 8, require boundary values on the left at $i=\{-4, \ldots, 0\}$ and right side at $i=\{N+1, \ldots, N+5\}$. On the left side the boundary values are provided using the reflective boundary condition combined with the sponge layer [11] to damp out the incoming and outgoing waves. On the right side, the boundary values are provided by the RANS model, and will be discussed later in this paper.

DOI: 10.1002/fld

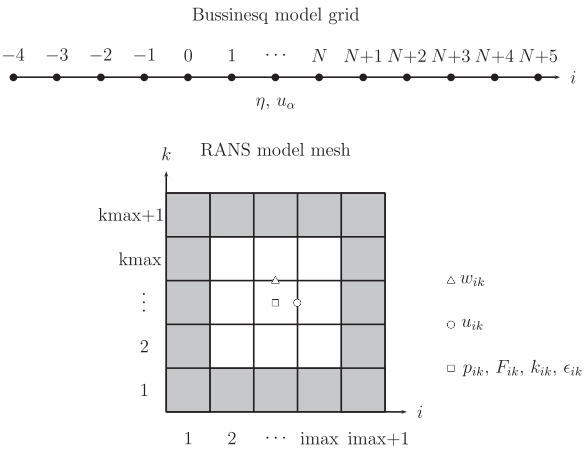

Figure 2. Boussinesq grid and RANS mesh.

3. RANS WAVE MODEL

3.1. Governing equation

The wave motion in the breaking zone (Figure 1) should ideally be modeled using the Navier–Stokes (N–S) equations. This direct numerical simulation, however, is very expensive, requiring a huge number of computational grid points to resolve turbulence length scales, and is generally beyond the capability of current computing technology for all but very small-scale studies [12]. A statistical approach is commonly employed to perform a turbulence simulation. Here, both the instantaneous velocity and pressure are split into the mean and fluctuating quantities: $u_i = \langle u_i \rangle + u_i'$[‡] and $p = \langle p \rangle + p'$. The continuity and momentum equations are then time-averaged, $1/(t_2 - t_1) \int_{t_1}^{t_2} (\ldots)\, \mathrm{d}t$, where $t_2 - t_1$ is long compared with that of the turbulent motion, to get the time-averaged RANS equations that govern the mean flow and pressure:

$$\frac{\partial \langle u_i \rangle}{\partial x_i} = 0 \tag{18}$$

$$\frac{\partial \langle u_i \rangle}{\partial t} + \langle u_j \rangle \frac{\partial \langle u_i \rangle}{\partial x_j} = -\frac{1}{\langle \rho \rangle} \frac{\partial \langle p \rangle}{\partial x_i} + g_i + \frac{1}{\langle \rho \rangle} \frac{\partial \langle \tau_{ij} \rangle}{\partial x_j} - \frac{\partial \langle u_i' u_j' \rangle}{\partial x_j} \tag{19}$$

where ρ is the density and τ_{ij} is the stress due to molecular viscosity. This integration produces the unknown correlation between the fluctuating velocities, the last term of (19), which represents the transport of momentum due to the fluctuating motion. To close (19), a nonlinear eddy viscosity model is used; here the $k-\varepsilon$ model is employed (see [3, 13] for details).

[‡]Since this is a 2-D problem in the x- and z-directions, the subscript i will have values 1 and 3 indicating both directions, respectively.

DOI: 10.1002/fld

3.2. Numerical solution

The 2-D RANS computational domain is discretized into a rectangular mesh as shown in Figure 2. This mesh employs the staggered grid system where the flow and pressure variables in a cell are located at different locations; the horizontal velocity, u, is located at the midpoint of the vertical edge, the vertical velocity, w, at the midpoint of the horizontal edge, and the pressure, p, F (volume of fluid), k, and ε in the center of the cell. Different shades in the mesh are used to denote the physical cells (white) and the ghost cells (gray) added for the boundary condition implementation.

The two-step projection method, as implemented in [14], is employed to solve (19), which is time-discretized as follows:

$$\frac{\tilde{u}_i^{n+1}-u_i^n}{\delta t}=-u_j^n\frac{\partial u_i^n}{\partial x_j}+g_i+\frac{\partial \tau_{ij}^n}{\partial x_j} \tag{20}$$

$$\frac{u_i^{n+1}-\tilde{u}_i^{n+1}}{\delta t}=-\frac{1}{\rho}\frac{\partial p_i^{n+1}}{\partial x_i} \tag{21}$$

For convenience, the $\langle\rangle$ are dropped and the sum of the molecular and turbulence stresses, the last two terms of (19), is simply written as τ_{ij}. In the first step, the projected velocity field $\tilde{u}_i^{n+1}$ is first calculated without taking the pressure p^{n+1} into account. In the second step, (21) and (18) are combined to get the pressure Poisson equation, which can be solved for the pressure p^{n+1}:

$$\frac{\partial}{\partial x_i}\left(\frac{1}{\rho^n}\frac{\partial p^{n+1}}{\partial x_i}\right)=\frac{1}{\delta t}\frac{\partial \tilde{u}_i^{n+1}}{\partial x_i} \tag{22}$$

With $\tilde{u}_i$ and p^{n+1} obtained from the previous steps, the solenoidal velocity field can be calculated from (21) and accordingly using the $k-\varepsilon$ equations, the kinetic energy k and the energy dissipation ε may be calculated.

The solenoidal velocity calculated in the two-step projection method is later used to determine the new volume of fluid distribution over the mesh via the advection equation:

$$\frac{\partial F}{\partial t}+u_i\frac{\partial F}{\partial x_i}=0 \tag{23}$$

Additional description of the volume of fluid method can be found in [14].

In summary, the computation in the RANS model is given in Algorithm 2. Suppose that in this algorithm initially we have all the dependent variables (u^n, w^n, p^n, F^n, k^n, and ε^n) in both the physical and ghost cells.

In Algorithm 2 all the variables along the boundary and in the ghost cells reflect the type of boundary conditions imposed on the associated sides of the domain. Relevant to the model coupling are the boundary values on the left side of the domain overlapping with the right side of the Boussinesq domain. On this side the inflow/outflow boundary condition, which requires the specification of the flow velocity, pressure, kinetic energy, and energy dissipation along the corresponding boundary/ghost cells, is imposed. While the specifications of the flow and turbulence

DOI: 10.1002/fld

Algorithm 2. Algorithm to numerically solve the RANS equations.

1. Calculate $\{\tilde{u}_{ik}^n, \tilde{w}_{ik}^n \,|i=2, \ldots, \text{imax}-1, k=2, \ldots, \text{kmax}-1\}$ from (20).
2. Solve (22) for $\left\{p_{ik}^{n+1} \,|i=2, \ldots, \text{imax}-1, k=2, \ldots, \text{kmax}-1\right\}$.
3. Calculate $\left\{u_{ik}^{n+1}, w_{ik}^{n+1} \,|i=2, \ldots, \text{imax}-1, k=2, \ldots, \text{kmax}-1\right\}$ from (21).
4. Solve the k–ε equations for $\left\{k_{ik}^{n+1}, \varepsilon_{ik}^{n+1} \,|i=2, \ldots, \text{imax}-1, k=2, \ldots, \text{kmax}-1\right\}$.
5. Calculate $\left\{F_{ik}^{n+1} \,|i=2, \ldots, \text{imax}-1, k=2, \ldots, \text{kmax}-1\right\}$ from (23).
6. Check if time step size, δt, satisfies the Courant (C_r) stability condition: $\delta t \leqslant \left|\frac{C_r \Delta x_i}{u_i}\right|$, $i=1, 3$, for all cells in the mesh. If $\delta t > \left|\frac{C_r \Delta x_i}{u_i}\right|$ in some cells, set it to the smallest $\left|\frac{C_r \Delta x_i}{u_i}\right|$.
7. Repeat the procedure until the end of the simulation.

boundary conditions are simply the imposition of the two variables on the boundary, this is not the case with the pressure boundary condition. The pressure along the boundary, which is required in step 2 of the algorithm, is given by the Neumann boundary condition $\nabla p \cdot \mathbf{n} = -1/\delta t(\mathbf{u}^{n+1} - \tilde{\mathbf{u}}) \cdot \mathbf{n}$, where $\mathbf{n}$ is the outward normal unit vector. It can be shown that with this boundary condition the linear system of equations resulting from the centered-difference spatial discretization of (22) in step 2 does not explicitly depend on p^{n+1} and $\tilde{u}$, but rather on $\mathbf{u}^{n+1} \cdot \mathbf{n}$ along the boundary (see, for instance, [15]). This simply means that the system of equations depends implicitly on p^{n+1} and $\tilde{u}$ through the Neumann boundary condition above. Therefore, solving the pressure Poisson equation does not require the imposition of the pressure along the boundary.

4. RATIONALE OF MODEL COUPLING

Suppose that the offshore-to-shoreline wave propagation as depicted in Figure 1 is to be simulated using the two models presented earlier, with the Boussinesq model being implemented in the nonbreaking zone and the RANS model in the breaking zone. To perform these computations, we should provide the boundary values on the side of each model located on the interface area. To do this, the two subdomains must overlap so that one model can provide the other with boundary values. Hence, from the computational point of view, the coupling of the Boussinesq and RANS models is possible provided that the data exchange scheme can be formulated, which will be explained in detail in the next sections.

In general and from the physics point of view, the two models cannot exchange appropriate boundary values because both are derived under different assumptions. For instance, in the Boussinesq model the flow is treated as irrotational and inviscid while in the RANS model it is rotational and viscous. Clearly the two different flow properties cannot be exchanged. Turbulence accounted for in the RANS is absent in the Boussinesq model, and therefore this property, which is one of the boundary values mentioned above, is impossible to exchange between the two models. However, in some situations certain properties of the physical problem may be so small that they can be neglected in the calculation without sacrificing accuracy. If for instance the interface of the two models is located in the nonbreaking zone and the wave under consideration satisfies the

DOI: 10.1002/fld

assumptions (weak dispersion and turbulence) made in the Boussinesq model, some quantities such as turbulence and rotationality in the RANS model become unimportant, and can be neglected. In such a case, data can be exchanged in a physically consistent context between the models. Therefore, due to the difference in physics and assumptions in the two models, the interface area must be located in the nonbreaking zone where the flow is relatively irrotational, the turbulence is small, and the wave under consideration must be relatively long with respect to the water depth.

5. COUPLING TECHNIQUE

The Boussinesq and RANS model algorithms have been presented in detail in the previous sections. To perform the calculations in each step of the algorithms, the interface boundary values should be provided. Coupling the two models, and hence algorithms, will provide these values. Since some steps require boundary values from the previous time steps and others from the current one, the two algorithms should be so connected that in proceeding from one step to the next the boundary values required in one model are always available from the other.

In Algorithm 1, the calculations in steps 1 to 3 require the boundary values from the previous time levels, which are given by the RANS model at the interface. In Algorithm 2, step 1 requires boundary values and variables given from the Boussinesq model at the interface from the previous time level. The second step of this algorithm requires u_{1k}^{n+1} along the interface boundary. This velocity can be obtained from the Boussinesq model, which calculates the new time level velocity in step 3. Once the second step is finished, RANS steps 3 to 5 can be computed since all of the computations in these steps are explicit. The corrector steps in the Boussinesq model can also now be calculated with the new time level boundary values obtained from the RANS model. To summarize, the hybrid computation is presented in Algorithm 3.

6. RANS BOUNDARY CONDITION

In Algorithm 3 the vertical profiles of velocities and the volume of fluid on the first column of the RANS model mesh are obtained from the Boussinesq model. These variables, however, are not immediately available from this model. To obtain these variables, certain transformations must be done with η and u_α. Figure 3 depicts the hybrid grid points on the interface. Shown in the figure are the first column of the RANS model mesh with 18 computational cells[§] and the grid points $N-2$, $N-1$, N, and $N+1$ of the Boussinesq model. Note that grid point N of the Boussinesq model is aligned with the right face of the first column in the RANS mesh.

To find the fluid distribution, F, in this column, we first determine η^* in the middle of the column at $x^*=x_N-\delta x/2$ by quadratically interpolating η_{N-2}, η_{N-1}, and η_N:

$$\eta^* = \frac{(x^*-x_{N-2})(x^*-x_{N-1})}{(x_N-x_{N-2})(x_N-x_{N-1})}\eta_N + \frac{(x^*-x_{N-2})(x^*-x_N)}{(x_{N-1}-x_{N-2})(x_{N-1}-x_N)}\eta_{N-1} + \frac{(x^*-x_{N-1})(x^*-x_N)}{(x_{N-2}-x_{N-1})(x_{N-2}-x_N)}\eta_{N-2}$$

[§]In a typical simulation, each column of the RANS model mesh may have tens or hundreds of computational cells.

DOI: 10.1002/fld

Algorithm 3. Hybrid algorithm.

1. **while** $t^n < t_{\text{end}}$ **do**
2. Calculate $\{\eta_i^{n+1} | i = 1, \ldots, N\}$ from (14).
3. Calculate $\{U_i^{n+1} | i = 1, \ldots, N\}$ from (15).
4. Solve the tridiagonal system of equations (13) for $\{u_{\alpha i}^{n+1} | i = 1, \ldots, N\}$.
5. Calculate $\{\tilde{u}_{ik}^n, \tilde{w}_{ik}^n | i = 2, \ldots, \text{imax}-1, k = 2, \ldots, \text{kmax}-1\}$ from (20).
6. Solve (22) for $\{p_{ik}^{n+1} | i = 2, \ldots, \text{imax}-1, k = 2, \ldots, \text{kmax}-1\}$. This requires u_{1k}^{n+1} from step 3.
7. Calculate $\{u_{ik}^{n+1}, w_{ik}^{n+1} | i = 2, \ldots, \text{imax}-1, k = 2, \ldots, \text{kmax}-1\}$ from (21).
8. Solve the k–ε equations for $\{k_{ik}^{n+1}, \varepsilon_{ik}^{n+1} | i = 2, \ldots, \text{imax}-1, k = 2, \ldots, \text{kmax}-1\}$.
9. Calculate $\{F_{ik}^{n+1} | i = 2, \ldots, \text{imax}-1, k = 2, \ldots, \text{kmax}-1\}$ from (23).
10. With the new η_i^{n+1} and $u_{\alpha i}^{n+1}$, calculate $\{E_i^{n+1} | i = 1, \ldots, N\}$ from (11). This requires $\{\eta_i^{n+1}, u_{\alpha i}^{n+1} | i = N+1, \ldots, N+5\}$.
11. Calculate $\{\eta_i^{n+1} | i = 1, \ldots, N\}$ from (16).
12. With the new η_i^{n+1} and $u_{\alpha i}^{n+1}$, calculate $\{F_i^{n+1} | i = 2, \ldots, N\}$ from (12). F_1^{n+1} is calculated based on the corresponding Boussinesq elevation.
13. Calculate $\{U_i^{n+1} | i = 1, \ldots, N\}$ from (17).
14. Solve the tridiagonal system of equations (13) for $\{u_{\alpha i}^{n+1} | i = 1, \ldots, N\}$.
15. Calculate

$$r_\eta = \frac{\sum_{i=1}^{N} |(\eta_i^{n+1})_{\text{old}} - (\eta_i^{n+1})_{\text{new}}|}{\sum_{i=1}^{N} |(\eta_i^{n+1})_{\text{old}}|}$$

and

$$r_u = \frac{\sum_{i=1}^{N} |(u_{\alpha i}^{n+1})_{\text{old}} - (u_{\alpha i}^{n+1})_{\text{new}}|}{\sum_{i=1}^{N} |(u_{\alpha i}^{n+1})_{\text{old}}|}$$

16. **if** $r_u < 10^{-4}$ and $r_\eta < 10^{-4}$ **then**
17. Repeat the procedure until the end of the simulation.
18. **else**
19. Goto step: 9.
20. **end if**
21. Check if time step size, δt, satisfies the Courant (C_r) stability condition: $\delta t \leqslant |\frac{C_r \Delta x_i}{u_i}|$, $i = 1, 3$, for all cells in the mesh. If $\delta t > |\frac{C_r \Delta x_i}{u_i}|$ in some cells, set it to the smallest $|\frac{C_r \Delta x_i}{u_i}|$.
22. **if** δt is changed **then**
23. Interpolate the new η^n, η^{n-1}, η^{n-2}, u_α^n, u_α^{n-1}, and u_α^{n-2} at all nodes on the Boussinesq grid.
24. **end if**
25. $t^{n+1} = t^n + \delta t$.
26. $n = n + 1$.
27. **end while**

DOI: 10.1002/fld

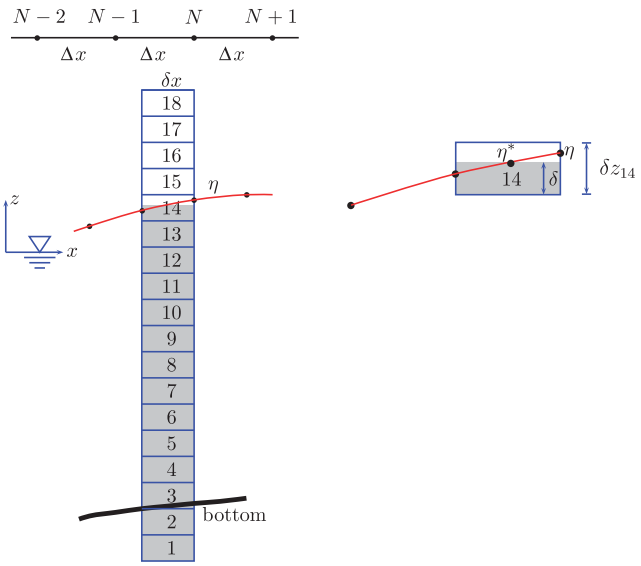

Figure 3. Calculation of the fluid distribution in column 1 of the RANS model mesh based on the Boussinesq model free surface elevation.

Then we search for the cell in which η^* is located in the column. In the exemplified figure this interpolated free surface elevation is located at cell 14. All the cells below it are full with $F=1$ and the cells above it are empty with $F=0$. For the free surface cell itself, F is calculated as the ratio of the height of the interpolated free surface elevation, δ, in the cell to the cell height, δz_{14}:

$$F_{14}=\frac{\delta}{\delta z_{14}} \tag{24}$$

Equations (7) and (8) are employed to calculate the vertical structures of u along the right face and w along the center of the first column of the RANS model mesh. Employing the second-order centered finite difference formula on (7) at x_N gives

$$u_N(z)=u_{\alpha N}-\frac{1}{2}(z^2-z_\alpha^2)_N\frac{u_{\alpha N+1}-2u_{\alpha N}+u_{\alpha N-1}}{\Delta x^2}$$
$$-(z-z_\alpha)_N\frac{(hu_\alpha)_{N+1}-2(hu_\alpha)_N+(hu_\alpha)_{N-1}}{\Delta x^2} \tag{25}$$

A similar scheme is employed to discretize (8) for the vertical velocity profile. As the reference velocity grid point is not aligned with the center of the RANS mesh where the vertical velocity is located, prior to the calculation, we first determine u_α at $x^*_{N+1}=x_{N+1}-\delta x/2$ and $x^*_{N-1}=x_{N-1}-\delta x/2$ by quadratic interpolation. Then the vertical structure of the velocity is calculated at $x^*_N=x_N-\delta x/2$:

$$w_{x^*_N}(z)=-z\frac{u_{\alpha x^*_{N+1}}-u_{\alpha x^*_{N-1}}}{2\Delta x}-\frac{(hu_\alpha)_{x^*_{N+1}}-(hu_\alpha)_{x^*_{N-1}}}{2\Delta x} \tag{26}$$

DOI: 10.1002/fld

The interface of the hybrid model will be located at the low turbulence area. In doing so, the turbulence variables required along the first column of the mesh are small, as will be demonstrated later in the model test simulation. Accordingly, transfer of turbulence variables from the Boussinesq to the RANS model, as done in the imposition of the flow boundary condition, is not necessary in the hybrid model. With this very weak assumption of turbulence at the interface, the employed condition at this location for turbulence quantities is zero horizontal gradient.

7. BOUSSINESQ BOUNDARY CONDITION

The specification of the Boussinesq boundary condition is the inverse procedure described in the previous section. In this procedure the reference velocity and the free surface elevation are calculated based on the velocity, u, and the fluid distribution, F, in the RANS model. This calculation is carried out on the five Boussinesq ghost grid points from $N+1$ to $N+5$. In this section we describe the calculation of the boundary condition on the grid point $N+1$. The identical procedure applies to the rest of the ghost grid points. Figure 4, a modification of Figure 3, depicts the RANS–Boussinesq model grid system on the interface area. As shown the two models employ different grid sizes. Without loss of generality, the Boussinesq grid is depicted here larger than the RANS model grid, which will be typically the case. To determine the free surface elevation and the reference velocity, we should first find the column in the RANS model mesh that aligns

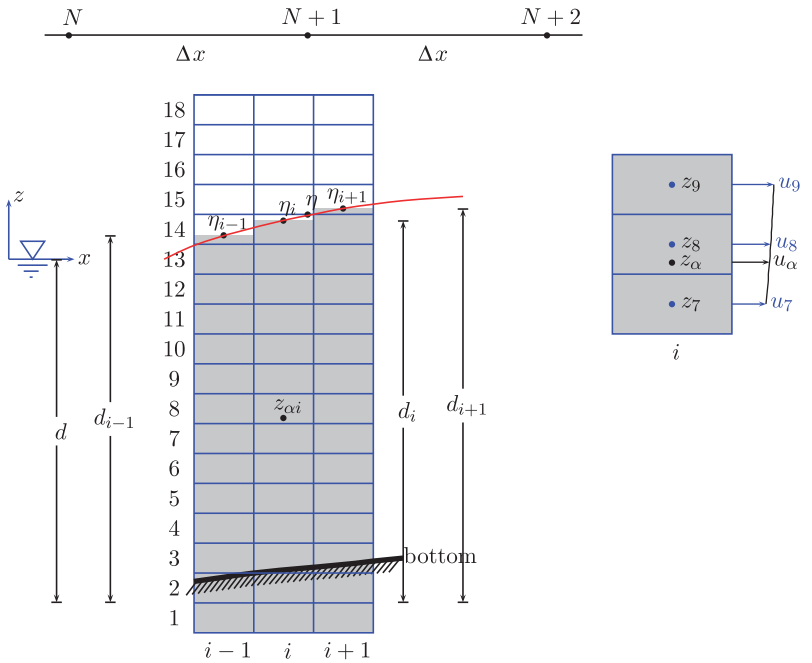

Figure 4. Calculation of the Boussinesq model free surface elevation and reference velocity based on the RANS model velocity and fluid distribution.

DOI: 10.1002/fld

with grid point $N+1$ of the Boussinesq model, i.e. column-i in Figure 4. Since in general the center of the RANS column does not perfectly align with the Boussinesq grid point, quadratic interpolation based on the RANS free surface elevations in column $i-1$, i, and $i+1$ is employed to determine the Boussinesq free surface elevation at grid point $N+1$. Note that, in the RANS model domain, the bottom of the lowest cell is assigned an elevation of 0. Accordingly, the RANS free surface elevation, as measured from the still water level, in column-i of the RANS model mesh is $\eta_i = d_i - d$, where d_i is the height of the water in column-i and d the height of the still water level (the same for all columns) in the RANS model mesh.

The reference velocity calculation starts with identifying the cell in the column of the RANS model mesh where the reference level is located. In Figure 4 the reference level in column-i, for instance, is located at cell 8. To determine the reference velocity to be given to the Boussinesq, a quadratic polynomial of the form

$$u_i = \beta + \theta(z - z_\alpha) + \gamma(z^2 - z_\alpha^2) \tag{27}$$

is fitted to the three neighboring velocities u_7, u_8, and u_9. Comparing (27) and (7), it is apparent that the reference velocity is the zeroth-order coefficient of the polynomial $u_{\alpha i} = \beta$. A similar procedure is employed to determine $u_{\alpha i-1}$ and $u_{\alpha i+1}$. Based on these three reference velocities, $u_{\alpha N+1}$ is calculated via interpolation. It is reiterated here that the Boussinesq model does not require any boundary values for vertical velocity or pressure. In the Boussinesq model, these are explicit functions of the horizontal velocity and free surface elevation.

8. HYBRID MODEL TEST

To test the model for both the validity and relative speedup, we use the hybrid model to simulate the following target scenarios:

(1) Solitary wave propagation.
(2) Standing wave motion.
(3) Sinusoidal wave overtopping of a seawall.
(4) Solitary wave overtopping of a levee.
(5) Hypothetical large-scale tsunami simulation.

In the first scenario, we consider the propagation of a solitary wave along a channel of constant depth, where as an important property, the wave height is constant as the wave propagates. This scenario is simulated using the hybrid model to see how well the model produces this property. In the second test, the hybrid model is used to generate a standing wave in a channel of constant depth. The wave is driven from one end of the channel toward a vertical, reflecting wall. This wave is reflected back toward the origin and thus two waves opposite in directions meet on the interface to create a standing wave. In the third simulation, the model is run to simulate the wave overtopping of a seawall. The numerical data collected in this simulation is the mass flux of the wave overtopping across the top of the structure. In the fourth simulation, we conduct the simulation of the solitary wave overtopping of a coastal structure. The calculated free surface elevation, before and after interaction with a levee, is recorded and compared with experimental data. In the last simulation, a hypothetical tsunami in deep water is generated. The hybrid model is employed to predict its evolution on the coast, including interaction with a breakwater.

DOI: 10.1002/fld

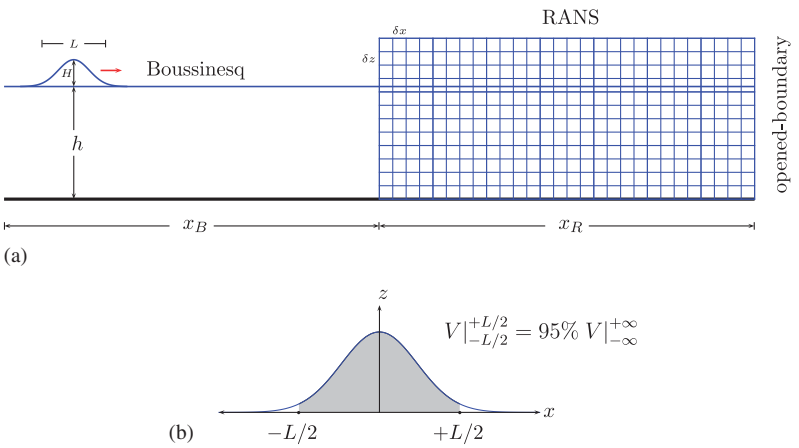

Figure 5. (a) Hybrid simulation setup for solitary wave propagation and (b) solitary wave form centered at the coordinate origin and symmetrical about the z axis. Gray area occupies 95% of the total water volume in the wave.

To measure the relative speedup of the hybrid computation, the first four scenarios are also run using the RANS model in the whole domain, a 'pure RANS model' simulation. The CPU times used by the pure RANS model to compute those scenarios are compared with the CPU times used in the hybrid model. This comparison gives the relative speedup of the hybrid model.

8.1. Solitary wave propagation

The first test simulates the propagation of a solitary wave in a 0.5 m deep and 100.0 m long channel. The domain is divided into two subdomains of equal lengths. The first 50.0 m subdomain, x_B, is occupied by the Boussinesq model and the second 50.0 m, x_C, by the RANS model. The wave is generated in the Boussinesq domain and propagates to the RANS model domain (Figure 5). To observe the behavior of the model with respect to the wave height variations, two waves of different nonlinearities, $\varepsilon = H/h$, and dispersiveness, $\delta = h/L$, are considered in the test. While a solitary wave is uniquely defined with only ε, the dispersive parameter is included here to both provide additional characterization of the wave and for use as a length scale. The solitary wave length, L, appearing in δ, is defined as the length of the symmetrical region in the solitary wave (gray-colored area in part (b) of Figure 5), which contains 95% of the total volume of water [16].

The initial free surface elevation and velocity of the solitary wave that are used to drive the model have the form

$$\eta = A_1 \operatorname{sech}^2[B(x - Ct)] + A_2 \operatorname{sech}^4[B(x - Ct)] \tag{28}$$

and

$$u = A \operatorname{sech}^2[B(x - Ct)] \tag{29}$$

The two equations are the analytical solution to the weakly nonlinear Boussinesq equation [8]. The derivation of the analytical solutions and the determination of the coefficients A, A_1, A_2, B, and C may be found in [17].

DOI: 10.1002/fld

In the following two solitary wave simulations, the fluid viscosity and the turbulence in the RANS model are set to zero. Hence both models are running under an inviscid (potential) flow condition, with no means in the RANS model to generate vorticity.

In the first solitary wave simulation, a small amplitude wave of height 0.05 m initially centered at $x=10.0$ m is used. The length of this wave in the 0.5 m deep channel is 6.64 m. The corresponding nonlinearity and dispersiveness are $\varepsilon=0.1$ and $\delta=0.075$, respectively. Both the Boussinesq and RANS models use uniform spatial grids. The Boussinesq grid size is $\Delta x=0.125$ m and the RANS grid sizes are $\delta x=0.0625$ m and $\delta z=0.0175$ m. In this simulation constant time step, $\delta t=0.01$ s, which corresponds to the Courant number $C_r=0.18$ in the Boussinesq model domain and $C_r=0.35$ in the RANS model domain, is employed. Here, the characteristic velocity used in the Courant number is the constant, linear long wave speed. To save some computational time, the RANS model starts its calculation when the free surface elevation of the Boussinesq model on the interface area (which acts as the boundary condition in RANS model) exceeds a threshold, $\eta_{\text{threshold}}=10^{-5}$ m. For this particular simulation, there are no calculations in the RANS model domain for the first 13.0 s of physical time or about 44.0% of the total time. In a huge computational domain, particularly for transient wave studies, this procedure can save a significant amount of computational time. This procedure is implemented for all the simulations in this study.

To see the evolution of the soliton as it propagates along the channel, several snapshots of the free surface elevation, η, at different times are captured at 6 s interval (Figure 6). The distance between two consecutive solitons is 13.9 m. The soliton moves at $13.9/6.0=2.32$ m/s. As the wave travels from the initial location to $x=80.0$ m, the wave height is invariant. The wave is correctly transmitted from the Boussinesq to the RANS domain. For comparison the simulation is also carried out using the Boussinesq model for the full domain $0\leqslant x\leqslant 100.0$ m. The wave profiles of this simulation are shown in dots in Figure 6. Both the hybrid and full-Boussinesq model wave profiles agree very well. Note the small, oscillatory tail following the soliton at later times in both the hybrid and full-Boussinesq results; this is a common occurrence when using the weakly nonlinear solitary wave solution in the fully nonlinear model [6].

In the second simulation, wave height is increased to 0.15 m. The wavelength is 4.3 m, shorter than the previous wave, and thus the wave steepness is considerably larger here. The nonlinearity is $\varepsilon=0.3$ and the dispersiveness is $\delta=0.12$. For this simulation the spatial and temporal grids are similar to the previous solitary wave simulation.

The snapshots of the soliton at 6.0 s temporal interval are presented in Figure 7. Here, the wave moves at a constant speed of 2.51 m/s, faster than the previous, smaller wave. The $t=18.0$ s soliton indicates that there is a smooth transition as the wave enters the RANS model domain from the Boussinesq model domain. As the wave travels in the RANS model domain, however, the wave height decays slowly as indicated by snapshots at $t=24.0$ s and $t=30.0$ s. From $t=24.0$ s to $t=30.0$ s, there is a 2.9% wave height reduction. Another simulation using the pure RANS model and employing the same wave and temporal/spatial grids confirms this decay (Figure 8). This figure obviously shows that the wave undergoes damping as it propagates from its initial position to $t=15.0$ s position. The wave at $t=9.0$ s, for instance, decays from 0.1366 to 0.1318 m at $t=15$ s, a 3.2% wave height reduction. As in the previous case, we also perform the full-Boussinesq simulation and the result is also presented in dots in Figure 7. The figure shows that the wave height of the full-Boussinesq run is invariant in the course of the simulation. From $t=18.0$ to 30.0 s, the soliton in the hybrid model, as a result of being smaller, lags slightly behind the full-Boussinesq wave, which is apparent in the close-up snapshot at $t=30.0$ s. The wave height decay is due to numerical dissipation in the RANS model.

DOI: 10.1002/fld

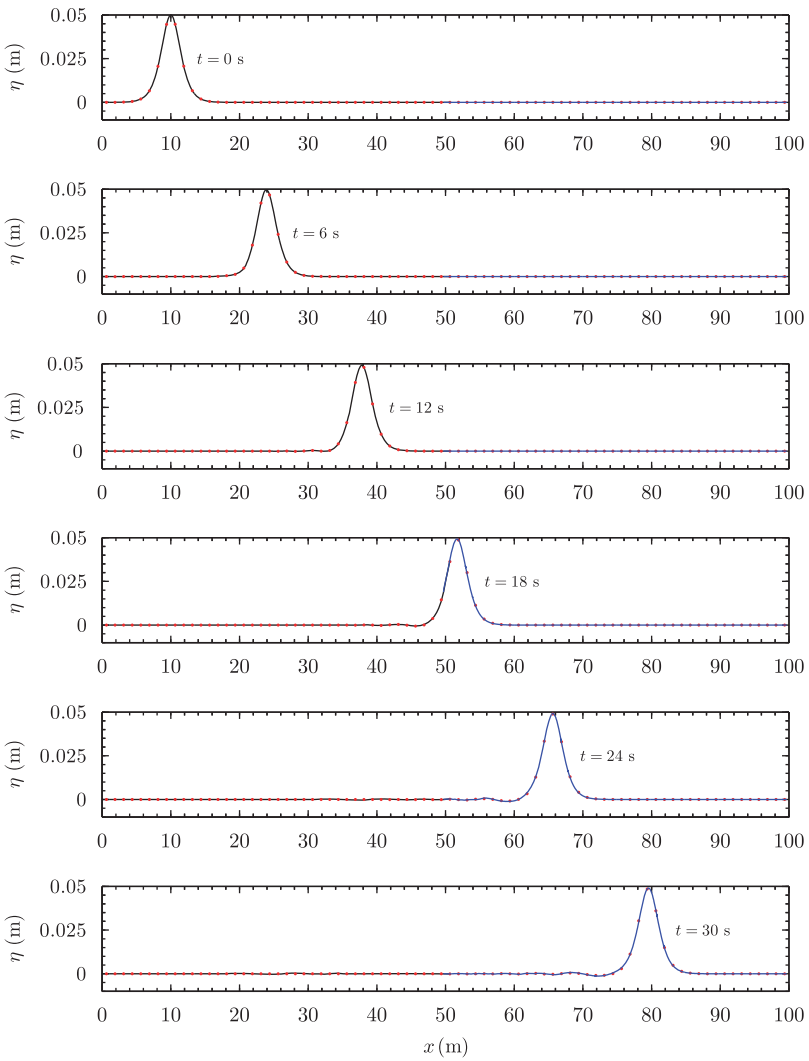

Figure 6. Solitary wave ($\varepsilon = 0.1$, $\delta = 0.075$) propagation in a 0.5 m deep channel simulated using the hybrid wave model. $0 \leqslant x \leqslant 50$ is Boussinesq model, $50 \leqslant x \leqslant 100$ is RANS model, and dots are the full-Boussinesq model.

Similar simulations using the RANS model on the whole domain are carried out to measure the relative speedups of the hybrid simulations. The temporal and spatial grids for the pure RANS simulations are similar to RANS model grids in the hybrid simulations. Hence, the total number of the RANS model horizontal computational grid points in these simulations is twice as many as in the hybrid model, while the number of vertical grid points remains unchanged. For the hybrid model, the computational clock time for the 0.05 m wave is 312.0 s and for the 0.15 m wave 317.0 s. The same simulations using the RANS model take 1260.0 and 1328.0 s for the 0.05 and 0.15 m waves, respectively. These computational clock times correspond to 30.4 s simulation and are run in

DOI: 10.1002/fld

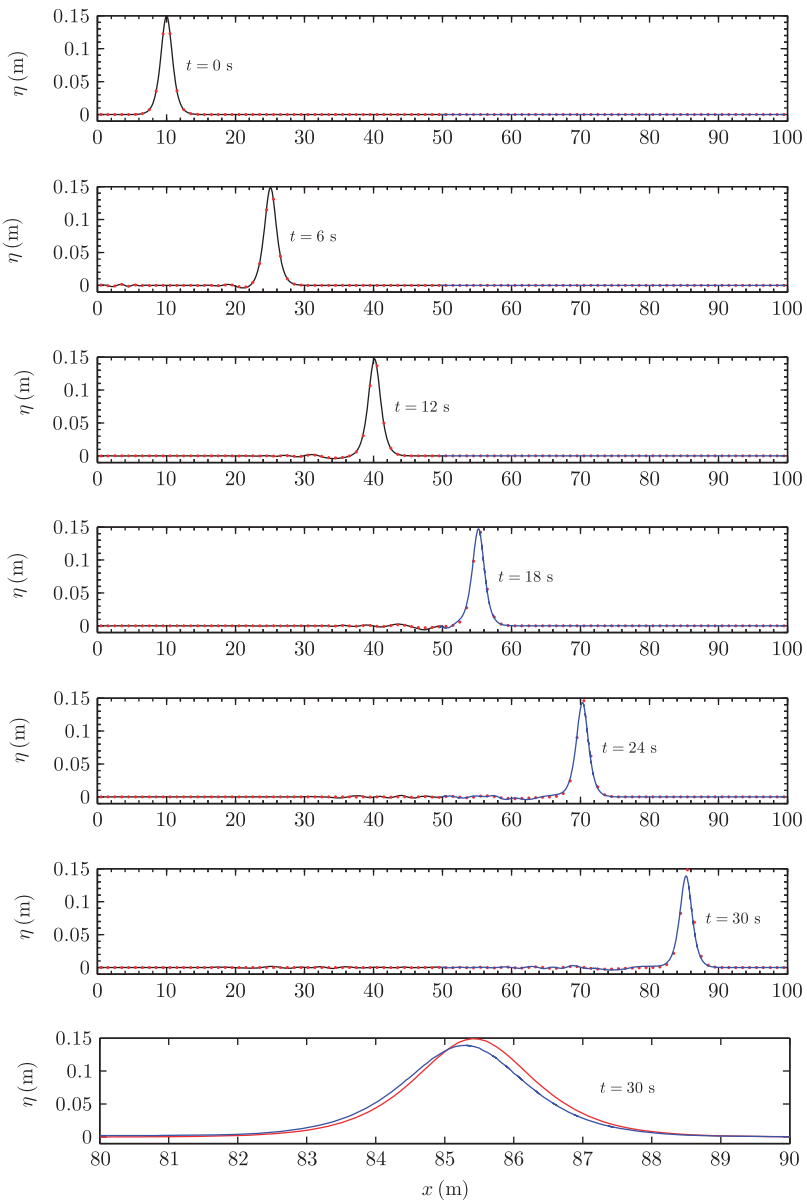

Figure 7. Solitary wave ($\varepsilon = 0.3,\ \delta = 0.12$) propagation in a 0.5 m deep channel simulated using the hybrid wave model. $0 \leqslant x \leqslant 50$ is Boussinesq model, $50 \leqslant x \leqslant 100$ is RANS model, and dots are the full-Boussinesq model.

a 3.0 GHz Pentium processor. The speedup factors gained by the hybrid model are 4.0 for the first wave and 4.2 for the second. In both the hybrid simulations the RANS-start threshold is reached at 13.4 s (44.0% of the total) simulation time and takes 25.0 s (8.0% of the total) computational

DOI: 10.1002/fld

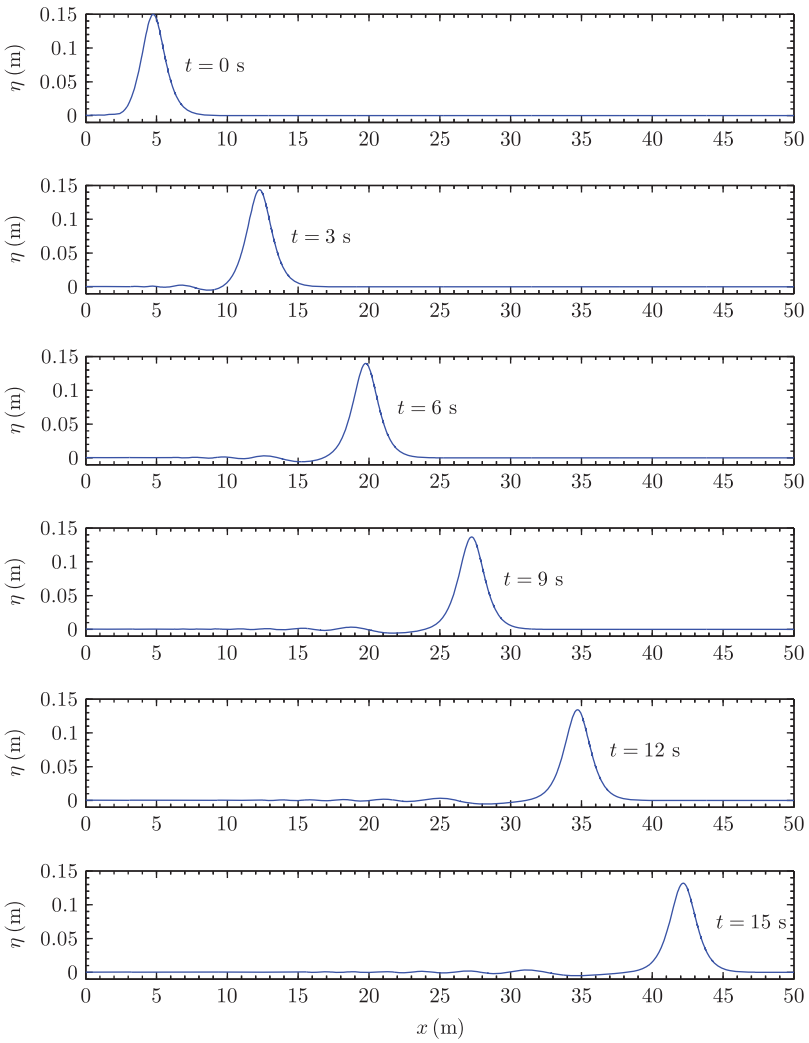

Figure 8. Solitary wave ($\varepsilon = 0.3$, $\delta = 0.12$) propagation in a 0.5 m deep channel simulated using the RANS model.

clock time. This comparison demonstrates that the threshold procedure can save, for these particular cases, 44.0% effort to run RANS in the hybrid model from the beginning.

8.2. *Standing wave motion*

In the following test, the hybrid wave model is used to simulate standing wave motion. The standing wave is created by superimposing two sinusoidal waves of the same height, H, moving in the opposite directions in a channel of constant depth, h, as shown in Figure 9. Here, the numerical channel is 72.0 m long, 0.5 m deep, and divided into two subdomains of equal lengths. The left

DOI: 10.1002/fld

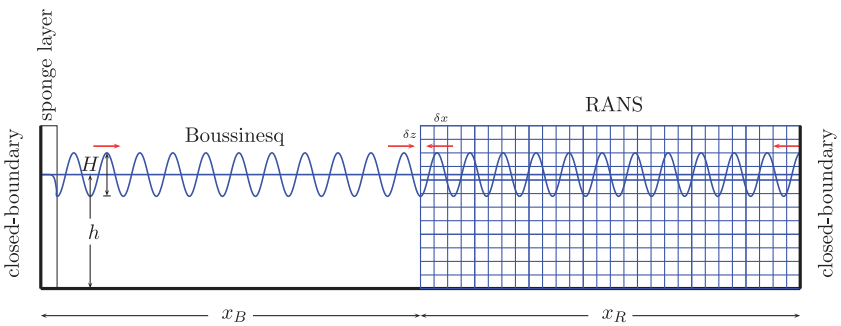

Figure 9. Hybrid model setup for standing wave simulation.

subdomain, x_B, is occupied by the Boussinesq model and the right one, x_R, by the RANS model. Both ends of the channel are impermeable walls and thus act as reflecting boundaries. Adjacent to the left boundary, a sponge layer is installed to damp out all the waves that enter the sponge layer area. For wave generation, we employ the internal source method [11]. Identical sinusoidal waves with height, H, propagate from the source to both the left and right directions. The wave that propagates to the left is damped out by the sponge layer and the one to the right will propagate along the Boussinesq and the RANS domains. As the wave reaches the right boundary in the RANS domain, it is reflected and propagates back toward the wave source and superimposes with the incoming wave to create the standing wave.

In the first test of this simulation, a relatively small wave, of 0.01 m in height with a period of 4.0 s, is generated 5.0 m to the right of the left boundary. A 3.0 m sponge layer is located adjacent to the left boundary as the damping mechanism. From the dispersion relationship, this wave has a wavelength of 8.7 m. The nonlinearity and the dispersiveness of this wave are then $\varepsilon = 0.02$ and $\delta = 0.06$, respectively. The Boussinesq domain is discretized into uniform grid, $\Delta x = 0.08$ m. Similarly, the RANS domain is uniformly discretized both horizontally, $\delta x = 0.04$ m, and vertically, $\delta z = 0.001$ m. The simulation is run for 100.0 s with the constant time step $\Delta t = 0.009$ s and zero viscosity and turbulence.

Figure 10 depicts the instantaneous wave profiles at six different instants. The first three profiles show the wave propagating to the right and reaching the wall at $t = 31.0$ s. Within this period the wave in the channel is not contaminated by the reflected wave from the right boundary and the height is still 0.01 m. Afterwards, the reflected wave starts propagating in the channel and is superimposed with the wave from the source. Since the two waves are identical and 180° out of phase, this superposition results in a standing wave in the channel with height 0.02 m, i.e. twice the original wave height. In all the snapshots the wave profile on the interface is smooth. Also, for comparison, the full-Boussinesq model simulation is carried out and the corresponding instantaneous profiles are plotted in the figure in dots. The two simulations are in good agreement.

Although the wave source input is linear (single harmonic), the wave undergoes some nonlinear evolution due to the nonlinear governing equations employed in both the Boussinesq and RANS models. The higher the nonlinearity, ε, the stronger the interaction. In the first case the nonlinearity is relativity small and its effect might not be so obvious in the wave profiles in Figure 10. Looking at the profile from a different perspective, for instance in the frequency domain, the nonlinearity effect becomes apparent. Figure 11 gives the amplitude spectrum of the time series of the free

DOI: 10.1002/fld

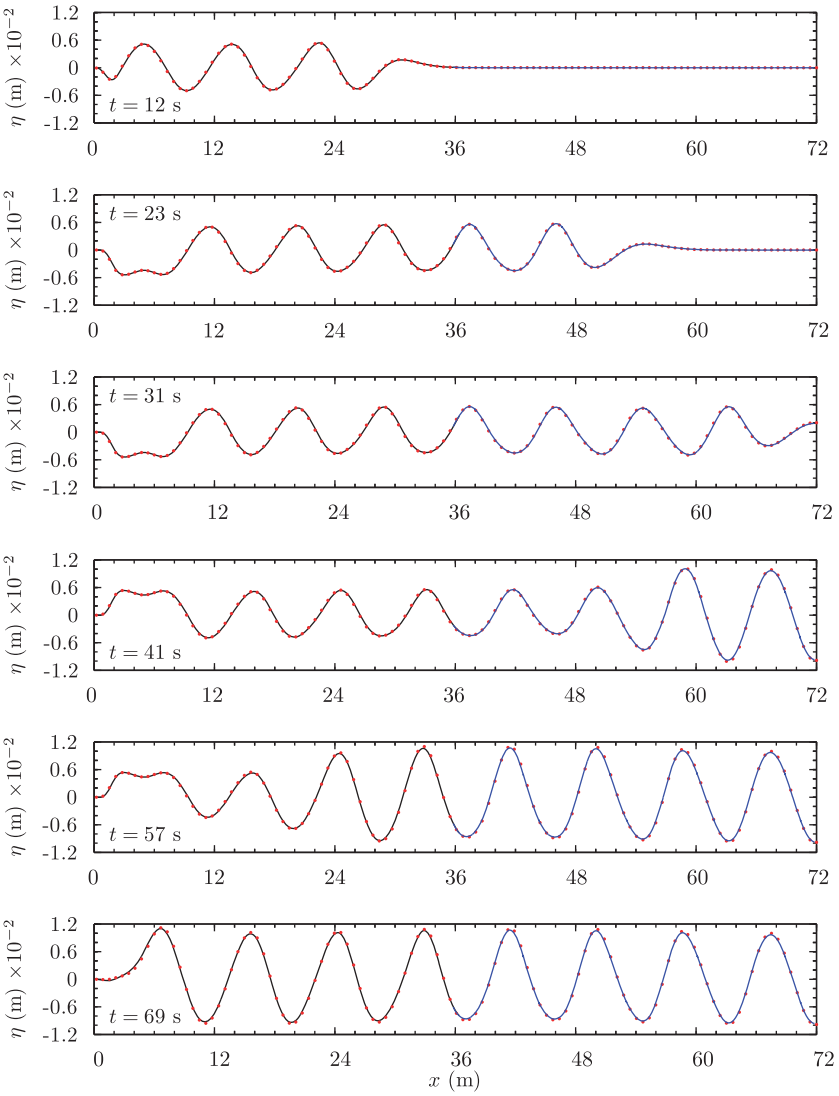

Figure 10. Standing wave ($\varepsilon = 0.02$, $\delta = 0.06$) motion in a 0.5 m deep channel simulated using the hybrid model. $0 \leqslant x \leqslant 36$ is Boussinesq model, $36 \leqslant x \leqslant 72$ is RANS model, and dots are the full-Boussinesq model.

surface elevation at $x = 38.0$ m. This particular time series is recorded before the reflected wave reaches the recording location. This spectrum clearly shows that the initially monochromatic wave, in the course of the propagation, transforms into a polychromatic wave. In the spectrum, there are two distinct spikes corresponding to $f_1 = 0.25$ Hz and $f_2 = 0.5$ Hz. Note that the frequency of the original signal is $f = 0.25$ Hz. Here, the first harmonic, f_1, interacts with itself resulting in the second harmonic, $f_2 = f_1 + f_1$. The interaction of the two harmonics, f_1 and f_2, is not too strong

DOI: 10.1002/fld

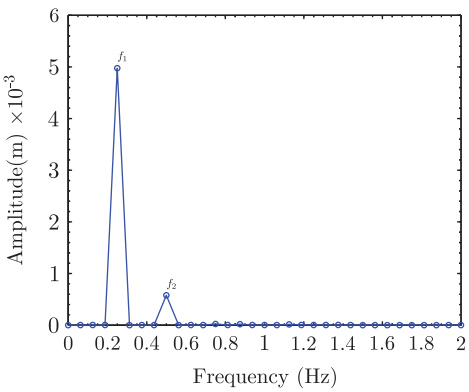

Figure 11. Amplitude spectrum of the free surface elevation time series of the $\varepsilon = 0.02$ standing wave simulation. The time series is recorded at $x = 38.0$ m before the reflected wave reaches this location.

in this case as there is no other distinct spike occurring in the spectrum. In the next test, as the wave height becomes higher, the effect of nonlinearity is stronger.

In the second test, the wave height is increased to 0.05 m, with a nonlinearity of $\varepsilon = 0.1$. The model setup remains the same as in the previous test. Figure 12 shows snapshots of the wave profiles as the wave propagates in the channel. Here, a smooth transition is again observed on the interface of the two models. Although there is very slight discrepancy between the hybrid and full-Boussinesq profiles, in general the two simulations show good agreement. The wave profile differs from the profile in the previous test in two ways: a wider trough and an occurrence of a secondary crest on the trough, which is due to the nonlinearity effect. The amplitude spectrum shown in Figure 13 has three distinct harmonics: f_1, f_2, and f_3. The first two harmonics are at the same frequencies as in the previous case. The interaction between the first and second harmonic results in the third harmonic whose frequency is $f_3 = f_1 + f_2 = 0.75$ Hz. This harmonic is obvious here, not in the previous test, since the amplitudes of the interacted harmonics, i.e. f_1 and f_2, are much higher in this test than the previous one.

In addition to the hybrid and full-Boussinesq simulations, we also simulate both cases of the standing wave motion using the RANS model for speedup comparison. The RANS model is applied on the whole domain with the same grid sizes as in the hybrid-RANS model. In the horizontal direction the number of computational grid points is twice as many as the grid points in the hybrid-RANS model mesh, and in the vertical direction both setups employ the same number of grid points. Therefore, in total, the number of grid points in the pure-RANS model is twice the number of grid points in the hybrid-RANS model. Both the pure-RANS and hybrid simulations are carried out in a 3.4 GHz Pentium processor for 400.0 s simulation time. In the first standing wave test, the hybrid model spends an average of 321.0 s for one wave period simulation while the pure-RANS model takes about 592.0 s. Hence, in the first test we gain a 1.8 factor of speedup. In the second test, for one wave period the hybrid and full-RANS models spend 248.0 and 592.0 s, respectively, and the gained speedup is 2.4.

8.3. Flux of sinusoidal wave overtopping

As reported in [18] the Beach Erosion Board (BEB) conducted a laboratory experiment in the Waterways Experiment Station of the Corps Engineers, at Vicksburg, Mississippi, to study wave

DOI: 10.1002/fld

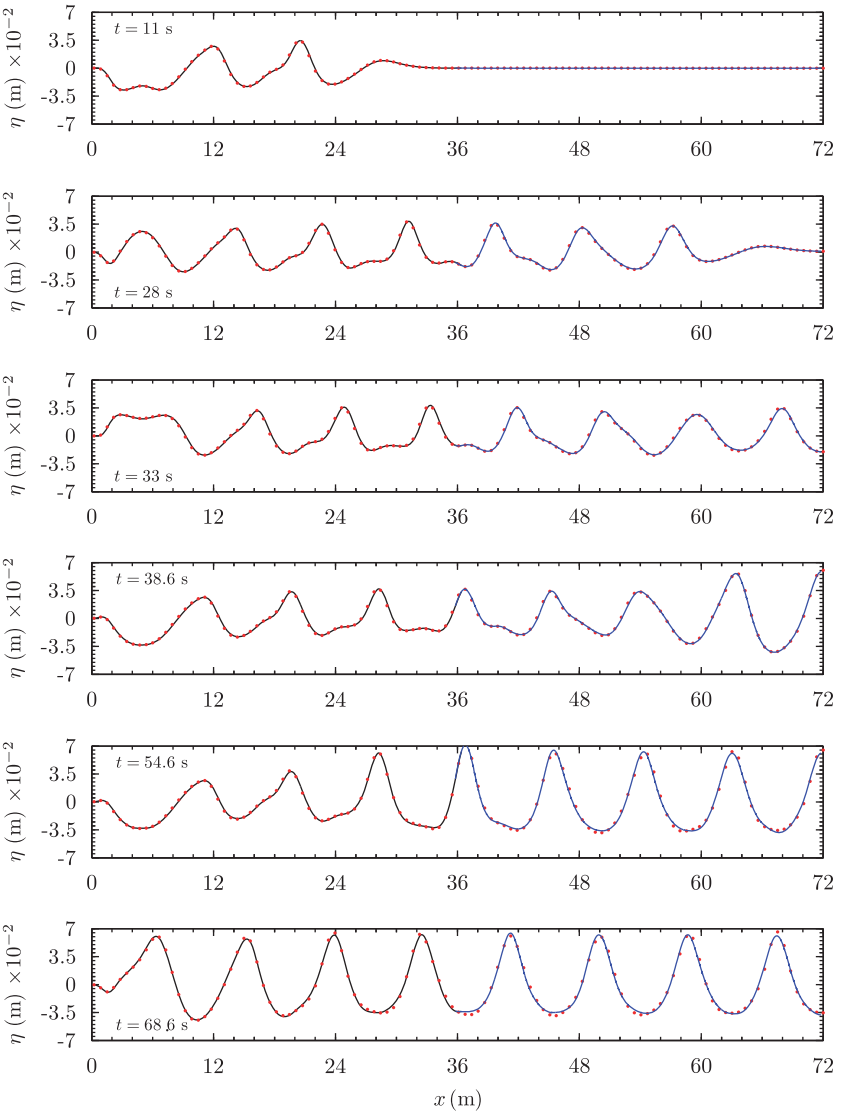

Figure 12. Standing wave ($\varepsilon = 0.1$, $\delta = 0.06$) propagation in a 0.5 m deep channel simulated using the hybrid model. $0 \leqslant x \leqslant 36$ is Boussinesq model, $36 \leqslant x \leqslant 72$ is RANS model, and dots are the full-Boussinesq model.

run-up and overtopping of shore structures. The experiment was conducted in a concrete wave flume 36.6 m long, 1.52 m wide, and 1.52 m deep. The model was an undistorted scale model with 1:17 length scale and 1:4.1 time and velocity scales. A wave maker was used on the upstream side of the flume for wave generation. Downstream of the flume, shore structures of various shapes (smooth slope, curve-faced wall, recurved wall, etc.) were built. Behind the structure a calibrated measuring tank was installed for collecting the overtopping water. The water from the first three

DOI: 10.1002/fld

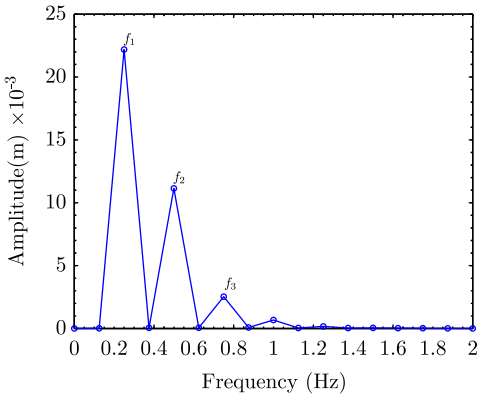

Figure 13. Amplitude spectrum of the free surface elevation time series of the $\varepsilon = 0.1$ standing wave simulation. The time series is recorded at $x = 38.0$ m and before the reflected wave reaches this location.

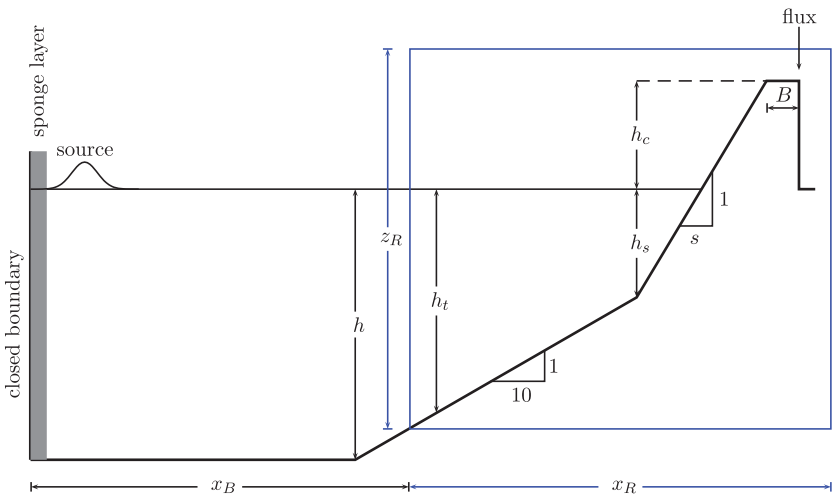

Figure 14. Experimental setup of the sinusoidal wave overtopping by the Beach Erosion Board; figure is not scaled.

or four waves was discarded to allow for the wave to attain a stable condition, after which the water from six or seven waves was collected in the tank for an overtopping flux measurement.

In this study, numerical simulations are undertaken of the BEB flux overtopping experiment using the hybrid model whose setup is given in Figure 14. For the simulation comparisons, the smooth structure data, as used by the previous researchers [19, 20] in their overtopping studies, are employed here. In the numerical simulations the wave is generated using the sinusoidal wave source combined with the sponge layer on the left boundary for damping. This approach is different from those used in [19, 20], where depending on the Ursell number, U_r, at the model boundary, h_t, Stokes or Cnoidal waves are used to drive the simulations, although the wavemaker in the physical

DOI: 10.1002/fld

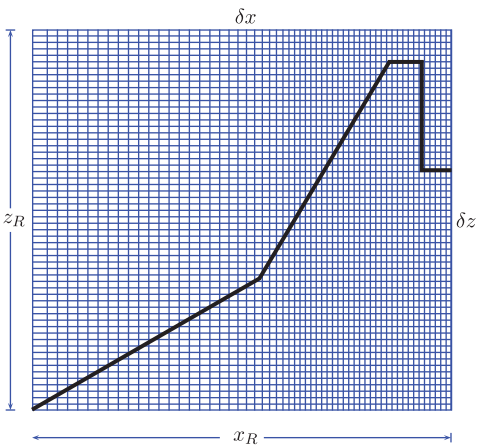

Figure 15. RANS computational mesh for hybrid simulation of BEB sinusoidal wave overtopping.

experiments created single harmonic waves only. In all the experiments, the structure with slope 1:s was fronted by a fixed 1:10 inclined floor. The domain is divided into the Boussinesq model subdomain, x_B, and RANS model subdomain, x_R.

The model interface is located near the toe of the 1:10 floor where turbulence is small and the wave does not yet break. In all the simulations, this interface divides the domain into the two subdomains with ratio $x_B/x_R \sim 7$. The height of the RANS domain, y_{R}, is large enough to prevent the overtopping wave from reaching the RANS model top boundary. For this simulation, the Boussinesq model domain is discretized into a uniform grid, Δx, and the RANS model domain is discretized, for efficiency, into nonuniform horizontal grid, δx, and uniform vertical grid, δz, as depicted in Figure 15. As shown in the figure, the mesh is relatively coarse near the left boundary and finer around the crest to allow for an accurate flux measurement. In addition for efficiency, a dynamic time step, δt, is employed in both the RANS and Boussinesq models; when the time step changes, free surface and velocity values at the previous time levels are interpolated or extrapolated to fit on a uniform time grid.

The flux is computed at the back edge of the structure and given by

$$q = \sum_{n=1}^{\mathrm{nt}} \sum_{k=1}^{k_{\max}} F_{ki}^{n} u_{ki}^{n} \delta z_k \delta t^n \tag{30}$$

The indices i and k correspond to a cell in the column along which the flux is computed. Since some cells might be not full, Equation (30) includes the corresponding value of the volume of fluid, F_{ki}^{n}. $n=1$ and nt are the time indices that correspond to the starting/ending times of the water collecting.

As in the BEB experiment, the simulation is run under various geometrical setups with different combinations of variables including the offshore depth, h, the depth at the toe of the structure, h_s, free board height, h_c, slope of the structure, s, wave height, and period. These variables, the experimental data, the computed hybrid model fluxes, and the published results from [19, 20] are presented in Table I. In general the computed fluxes are in good agreement (slight difference) with the experimental data and consistent with the results of the previous two researchers.

DOI: 10.1002/fld

Table I. Experimental and simulated fluxes of the BEB sinusoidal wave overtopping.

Run*	h (m)	h_s (m)	h_c (m)	H (m)	T (s)	$q^{\dagger}_{\text{data}}$ (m^2/s)	q_{KW} (m^2/s)	q_{OTT} (m^2/s)	q_{hyb} (m^2/s)
1	0.529	0.081	0.054	0.107	1.549	0.0073	0.0030	0.0039	0.0032
2	0.529	0.081	0.107	0.107	1.549	0.0004	0.0003	0.0007	0.0003
3	0.609	0.161	0.054	0.107	1.549	0.0071	0.0058	0.0066	0.0075
4	0.609	0.161	0.107	0.107	1.549	0.0040	0.0015	0.0019	0.0055
5	0.609	0.161	0.054	0.081	1.858	0.0065	0.0058	0.0062	0.0062
6	0.529	0.081	0.054	0.107	2.616	0.0066	0.0060	0.0074	0.0071
7	0.529	0.081	0.107	0.107	2.616	0.0019	0.0018	0.0025	0.0030
8	0.529	0.081	0.161	0.107	2.616	0.0044	0.0002	0.0007	0.0012
9	0.609	0.161	0.054	0.107	2.616	0.0104	0.0100	0.0118	0.0128
10	0.609	0.161	0.107	0.107	2.616	0.0044	0.0050	0.0064	0.0069
11	0.609	0.161	0.161	0.107	2.616	0.0009	0.0018	0.0028	0.0024
12	0.529	0.081	0.054	0.081	3.634	0.0065	0.0070	0.0076	0.0063
13	0.609	0.161	0.054	0.081	3.634	0.0093	0.0081	0.0086	0.0086
14	0.609	0.161	0.107	0.081	3.634	0.0055	0.0037	0.0044	0.0026
15	0.609	0.161	0.161	0.081	3.634	0.0018	0.0011	0.0016	0.0015
16	0.609	0.161	0.215	0.081	3.634	0.0008	0.0011	0.0002	0.0004
17	0.529	0.081	0.054	0.107	2.616	0.0054	0.0073	0.0069	0.0064
18	0.529	0.081	0.161	0.107	2.616	0.0014	0.0009	0.0008	0.0016
19	0.448	0.000	0.054	0.107	2.616	0.0043	0.0044	0.0041	0.0030
20	0.448	0.000	0.107	0.107	2.616	0.0022	0.0008	0.0009	0.0004

*Runs 1–16 use $s=3$ and runs 17–20 use $s=1.5$.
†Flux is presented in dimensional form instead of dimensionless form as in [19].

The slight discrepancies in these results are attributed to the small differences in the modeled physics and numerical accuracy in the non-breaking part of the domain. This datum also demonstrates the relatively wide variability that can be found in published overtopping predictions, particularly for low overtopping rates.

As previously explained, the interface of the model should be located such that the turbulence intensity on the interface is small. To provide insight into this, in Figure 16 the instantaneous intensity of the turbulent kinetic energy, k, for run 1, is given. This figure shows that the intensity of the turbulent kinetic energy, k, is high near the structure compared with other locations. The kinetic energy near the structure is roughly $10^{-2}\,m^2$ in contrast to a value less than $10^{-2}\,m^2$ on the interface. As the wave approaches the structure, the wave height to depth ratio becomes so large that the wave breaks while impinging on the structure and hence releases kinetic energy. Figure 16 also indicates that in the course of the simulation the location of the turbulence hot spot remains close to the crest of the structure while the interface area is always low in kinetic energy. This satisfies the requirement that the turbulence intensity should be low on the interface.

To benchmark the simulation time, five of the previous simulations are re-run using the RANS model in the whole domain. Table II presents the run times per wave period for both the hybrid and pure-RANS wave models for the five selected runs. In discretizing the pure-RANS model, the part of the hybrid domain where the RANS model is applied uses an identical mesh as in the hybrid case. Offshore of this point, where the Boussinesq model is used in the hybrid model, the pure-RANS model domain is uniformly discretized with a grid size equal to the grid size at the hybrid interface location.

DOI: 10.1002/fld

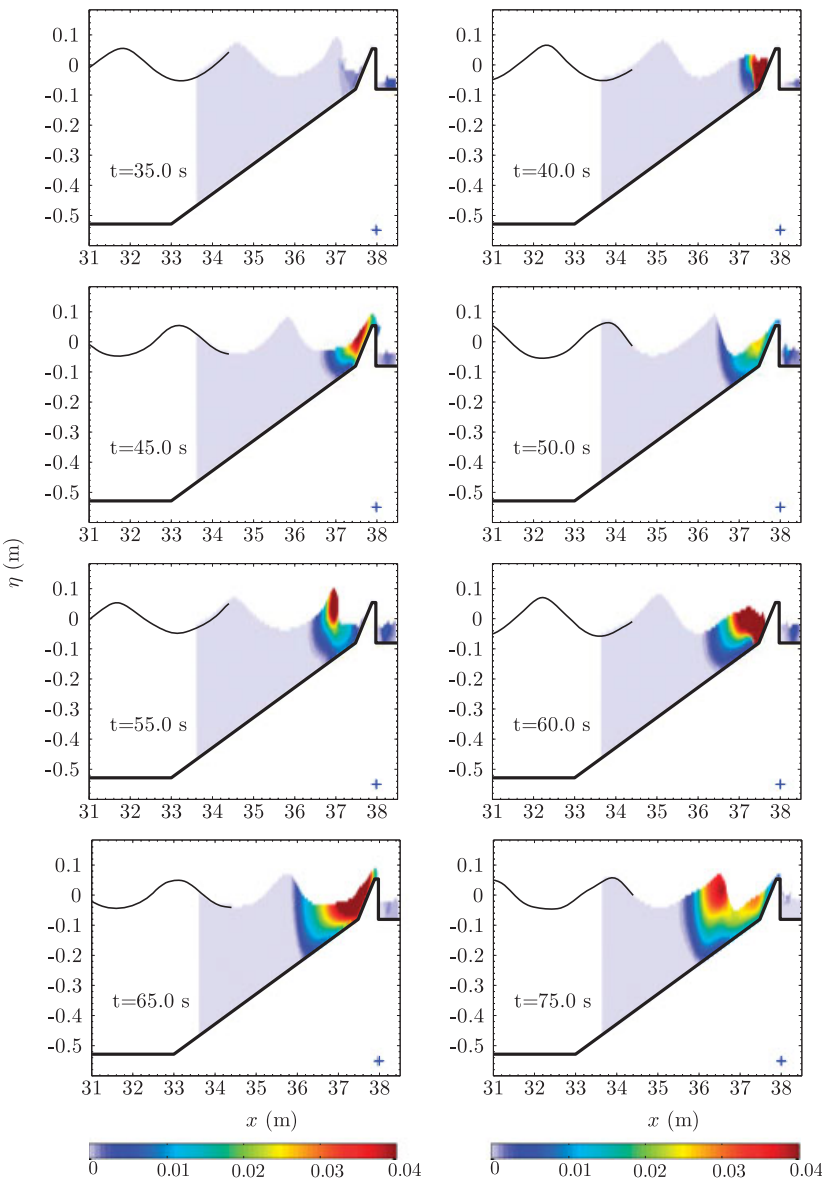

Figure 16. Turbulence kinetic energy distribution of the BEB sinusoidal wave overtopping, run 1.

The computational times of both the hybrid and pure-RANS simulations are summarized in Table II. From this table it is clear that the speedup due to use of the hybrid model is significant, ranging from a factor of near 10.0 (run 18) to over 17.0 (run 3 and 5). This large speedup is of course due to the smaller RANS mesh used in the hybrid model. However this difference is two fold: a single iteration of the Poisson pressure solver requires less time with a smaller matrix and a smaller matrix will converge in fewer iterations of the Poisson solver.

DOI: 10.1002/fld

Table II. Computation time per wave period of the BEB sinusoidal wave overtopping, runs 3, 5, 9, 18, and 19. A_H and A_R are the 'areas', the product of the number of grid points in the x- and z-directions, $nx \times nz$, of the meshes employed in the hybrid and pure RANS models.

	Hybrid			RANS				
Run	$nx \times nz$ (A_H)	t (s)	Flux (m^2/s)	$nx \times nz$ (A_R)	t (s)	Flux (m^2/sec)	A_R/A_H	Speedup
3	118×72	16.7	0.0075	798×86	269.8	0.0075	8.1	16.2
5	118×82	26.2	0.0062	798×98	454.7	0.0061	8.1	17.4
9	262×76	85.6	0.0128	826×122	1259.2	0.0122	5.1	14.7
18	149×72	25.5	0.0016	845×72	242.6	0.0010	5.7	9.5
19	115×52	10.5	0.0030	811×56	120.8	0.0032	7.6	11.5

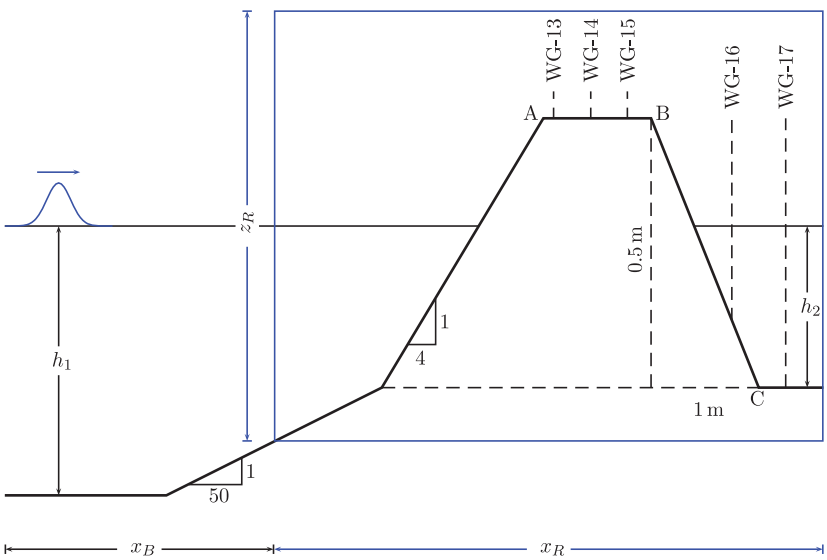

Figure 17. Experimental setup of the HR solitary wave overtopping.

8.4. *Free surface elevation of solitary wave overtopping*

In 1996, Hydraulic Research (HR) Wallingford in the U.K. performed an experiment on solitary wave overtopping of a breakwater. The experimental setup is given in [20] and depicted here in Figure 17. The wave flume used in this experiment was 40.0 m long and 0.5 m wide and filled with water to $h_1 = 0.7$ or 0.6 m seaward of the breakwater and $h_2 = 0.3$ or 0.2 m behind the breakwater. A breakwater with 1:4 or 1:2 slope was built at the right end of the flume. This breakwater, which is 0.5 m high and 0.16 m wide on the top, was fronted by a 1:50 inclined floor. To measure the free surface elevation of the overtopping water, a series of wave gauges were installed on top of and behind the breakwater. The first gauge, WG-13, was located 0.015 m behind the leading edge (A), the second, WG-14, and third, WG-15, gauges were installed 0.055 and 0.11 m from the first gage, respectively. Depending on whether the first or second depth was used, the fourth gage,

DOI: 10.1002/fld

Table III. Wave height and water depth of the HR solitary wave overtopping.

Test	H (m)	h_1 (m)	h_2 (m)	WG-16* (m)
4c7a	0.07	0.7	0.3	0.72
4c7b	0.10	0.7	0.3	0.72
4c7c	0.12	0.7	0.3	0.72
4c6a	0.07	0.6	0.2	1.10
4c6b	0.10	0.6	0.2	1.10
4c6c	0.12	0.6	0.2	1.10
4c6d	0.15	0.6	0.2	1.10

*Distance from the backedge B in Figure 17.

WG-16, was located 0.72 or 1.1 m behind the back edge (B) of the breakwater. The last gauge was fixed 0.44 m behind the back toe (C) of the breakwater. The experiment was conducted for several solitary wave heights and water depths as given in Table III.

The simulation setup is very similar to the one used in the BEB simulation where the uniform Boussinesq grid is coupled with the RANS nonuniform x- and uniform z-grids. The interface divides the domain into two segments, x_B and x_R, with ratio $x_B/x_R \sim 9$. A dynamic time step is also employed in the simulation. In all tests the initial location of the solitary wave is 10.0 m from the left boundary of the Boussinesq domain.

The time series of the free surface elevation of the hybrid simulations and the experimental data are presented in Figure 18. For comparison the numerical simulation data from [20] (called OTT), using a nonlinear shallow water wave equation model, are also given in the same figures. In all the simulations, the hybrid wave model shows a clear bias toward overpredicting the water elevation on top of the structure. This is consistent with the OTT simulations, which show an even larger bias. On the lee side of the breakwater, the hybrid simulations in general agree quite well with the data, with a remarked improvement over the shallow water equation-based OTT.

The relative speedup is also measured for this case. The discretization of the pure-RANS domain is almost the same as in the previous BEB discretization. The only difference is that to save computational time in the pure-RANS simulations, the domain offshore of the interface location of the hybrid model is discretized nonuniformly, with an increasingly coarse grid in the deeper water. The computation times of both the hybrid and pure-RANS simulations are summarized in Table IV. As with the speedups in the BEB tests, the reduction in CPU time shows a factor greater than the decrease in RANS domain size. This, again, is due to the Poisson solver requiring fewer iterations to converge with a smaller matrix size.

8.5. Hypothetical tsunami simulation

In many tsunami events, such as the devastating tsunami that struck the West Sumatera Coast in 2004, the wave is generated by a sudden uplift or subsidence of the seafloor following a massive tectonic earthquake. The vertical displacement of the seafloor disturbs the equilibrium of the water column above it and in consequence the water mass spreads as a long wave to attain a new gravitational equilibrium.

In the last model application, the hybrid wave model is used to simulate the propagation of a tsunami-type wave from the open ocean to the coast. The simulation setup is given in Figure 19. The domain consists of a 1.0 km deep ocean connected with a 1:50 seafloor. A 4.0 m elevation

DOI: 10.1002/fld

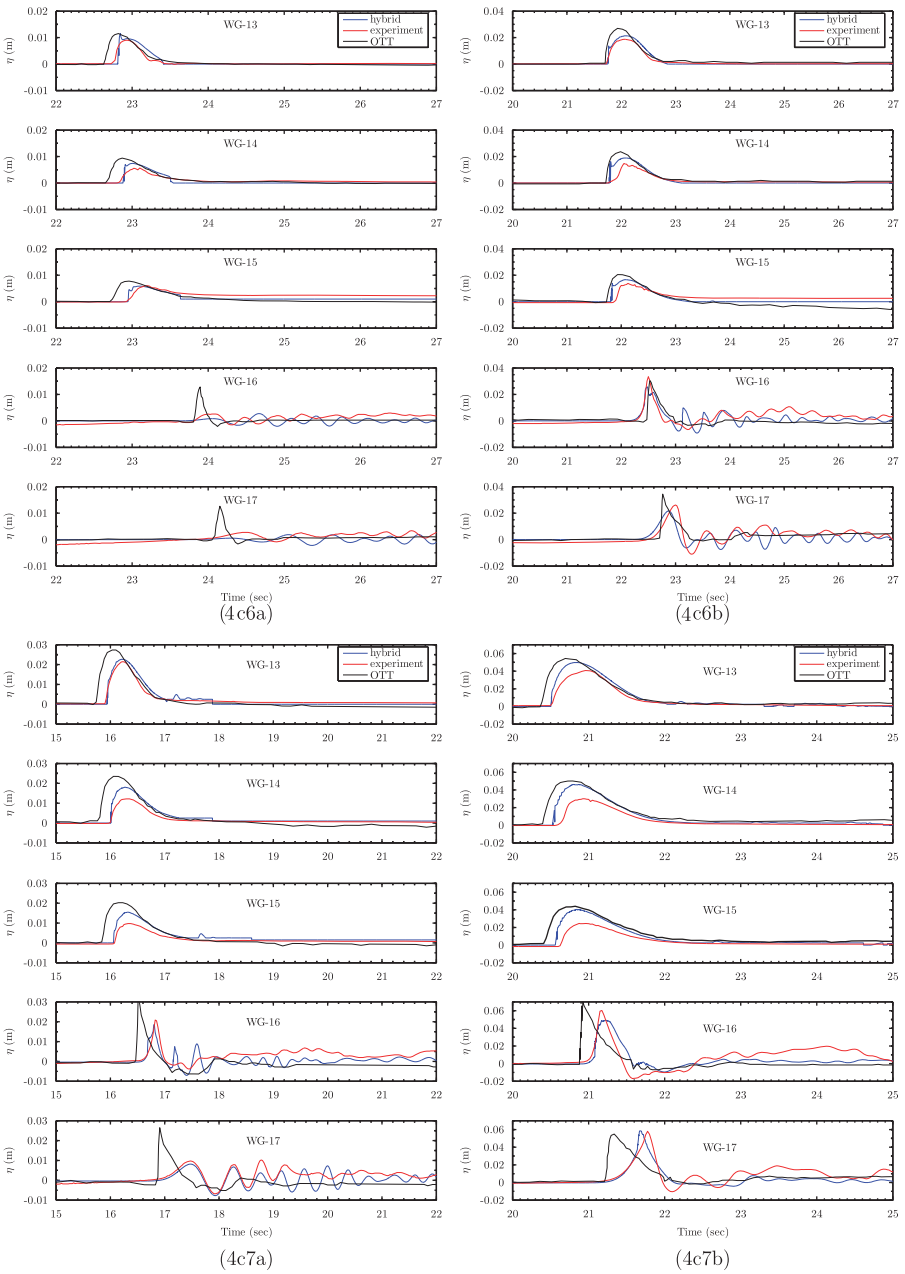

Figure 18. Time series of the free surface elevation of the HR solitary wave overtopping.

breakwater is placed along the coast in a shallow water depth of approximately 3 m. For the hybrid simulation, the Boussinesq model occupies 93.0% of the total horizontal domain length; the RANS model occupies only 7.0%, which is located in the nearshore region. The domains

DOI: 10.1002/fld

Table IV. Computation time of HR solitary wave overtopping.

	Hybrid		RANS			
Test	$nx \times ny$ (A_H)	t (s)	$nx \times ny$ (A_R)	t (s)	A_R/A_H	Speedup
4c7a	303×122	746	610×202	3605	3.3	4.8
4c7b	419×122	1613	765×202	7580	3.0	4.7
4c7c	461×122	2494	862×202	9930	3.1	4.0
4c6a	395×142	2236	746×202	9134	2.7	4.1
4c6b	435×122	2367	821×202	12 303	3.1	5.2
4c6c	435×122	2518	821×202	15 126	3.1	6.0
4c6d	435×122	2573	821×202	10 951	3.1	4.3

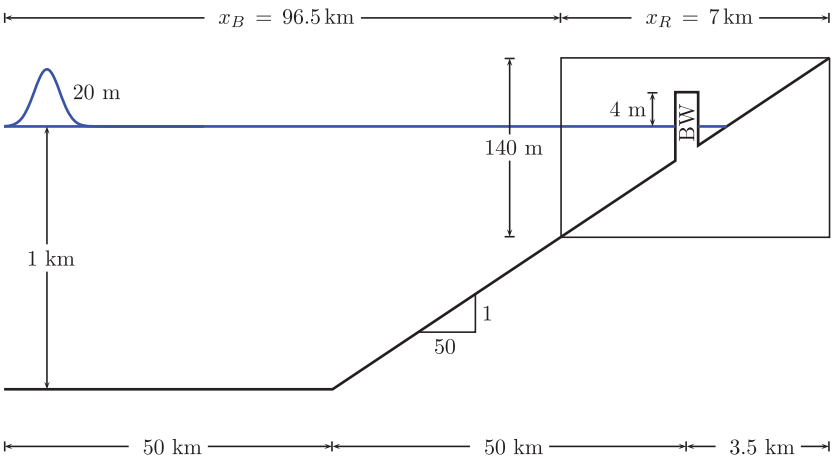

Figure 19. Simulation setup of the hypothetical tsunami generation and propagation. Figure is not scaled.

of the Boussinesq and RANS models are uniformly discretized into $\Delta x = 15.0$ m and $\delta x = 4.0$ m and $\delta z = 0.5$ m grids, respectively. The simulation is run with a dynamic time step for 1500.0 s of simulation time. The wave is generated 100.0 km offshore by uplifting the sea surface to form a Gaussian shape wave with zero initial velocity. The generated wave has an offshore wavelength of roughly 10.0 km, and thus might represent the leading wave of a leading elevation tsunami.

Figure 20 shows snapshots of the wave at three different times. At $t = 1070.0$ s the wave has just reached the detached breakwater with a turbulent wave front that appears as an 18.0 m high wall of water moving at a speed of nearly 10 m/s. The second snapshot in Figure 21 gives the wave at $t = 1140.0$ s. Just 70.0 s after the wave reaches the breakwater, the coast is flooded up to 1.6 km inland. The average flow depth in the flooded area is 13.0 m, with a 16.0 m/s average speed. The bottom part of the figure shows the detail of the flow around the breakwater. At the upstream side the flow moves 6.0 m/s and due to the breakwater acting as a sill, the velocity increases nearly three times to 17.0 m/s at the leeward side of the breakwater. In addition note the regions of separation at the breakwater corners, indicated by a relatively low fluid speed. Figure 22 shows

DOI: 10.1002/fld

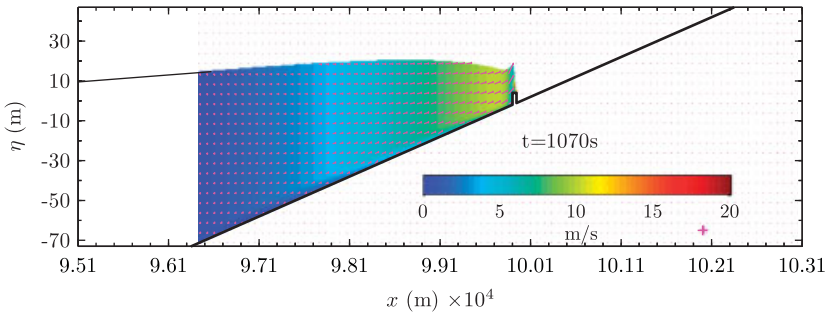

Figure 20. Snapshot of tsunami wave reaching the breakwater.

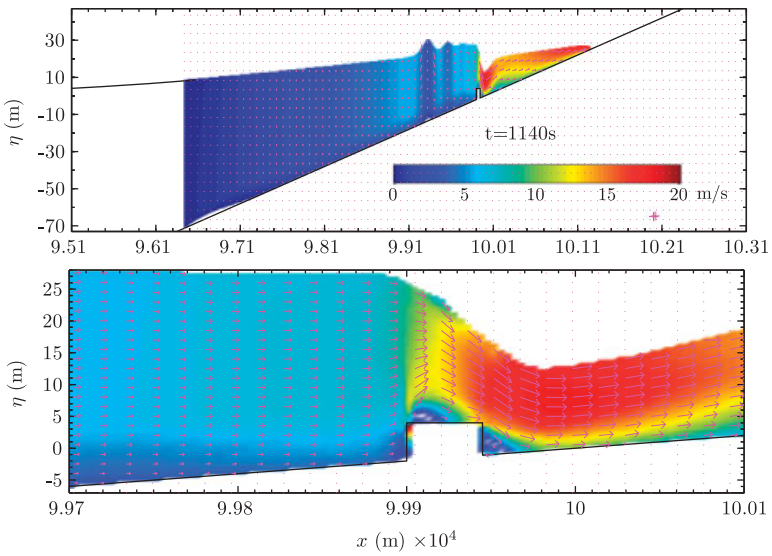

Figure 21. (Top) Snapshot of tsunami wave passing the detached breakwater and (bottom) close-up look of velocity near the breakwater.

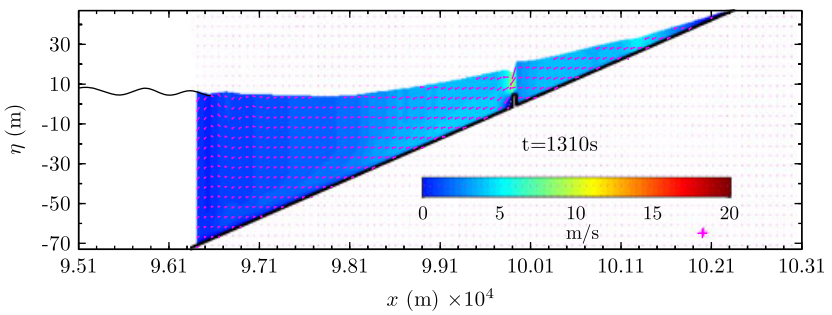

Figure 22. Snapshot of tsunami wave inundating the coast at the time of maximum run-up.

DOI: 10.1002/fld

the tsunami at maximum run-up. At this stage, about 4.0 min after the wave front first reaches the breakwater, the water has inundated 2.3 km of the coastal area.

For this model test, a time benchmark is not done as in the previous model tests. With current computing limitations, it is not practically possible to run the whole simulation with the RANS model without some type of parallel implementation of the model, which is not the focus here. Thus, this example demonstrates the potential of the hybrid model to routinely tackle multi-scale problems that would otherwise require significant computational capacity.

9. CONCLUSION

A hybrid wave model is developed that two-way couples numerical models based on the 1HD inviscid Boussinesq-equations and the 2-D viscous turbulence-closed RANS equations. The hybrid model is intended for large-scale wave simulation, which from either the accuracy or computational point of view is not possible to carry out using either model alone. The Boussinesq model will typically solve a large spatial portion of the computational domain, from the wave generation area to the pre-breaking zone with good accuracy and minor CPU needs. Coupled with the RANS model, turbulence and breaking waves in the nearshore can be simulated with high accuracy.

The model tests suggest that the current hybrid model is able to perform a broad range of nonbreaking/breaking wave tasks, from small-scale analytical and laboratory experimental scenarios to large-scale tsunami simulation, with good accuracy and efficient computational time. The primary deficiency of the presented model is that the location of the interface must be specified *a priori*. As mentioned, this location should be situated where turbulence is very low, such that both models are correctly describing the local physics. Ideally, the interface would be dynamically located, and allowed to move either landward or seaward as the local conditions dictate. The implementation of such a dynamic interface relies more on coding flexibility than the implementation of a correct physical boundary condition, as is the focus of this paper, and is left as a future enhancement to be incorporated into the coupling presented here.

For future studies, the hybrid algorithm herein can be readily employed to extend the model coupling to the 2HD Boussinesq and 3-D RANS wave models, as the numerical algorithms of the higher-dimension models remain the same. For large-scale simulation with detailed turbulence computation inside the breaker zone, the computational time is very high. This is of course due to the huge number of computational grid points employed in the finer RANS mesh. Integrating a parallelized RANS solution scheme into the hybrid model could greatly reduce the wall clock time, and may further facilitate the regular use of the hybrid and RANS models by engineers and scientists.

REFERENCES

1. Kennedy AB, Chen Q, Kirby JT, Dalrymple RA. Boussinesq modeling of wave transformation, breaking, and runup I: 1D. *Journal of Waterway, Port, Coastal, and Ocean Engineering* 2000; **126**(1):39–47.
2. Chen Q, Kirby JT, Darlymple RA, Kennedy AB, Chawla A. Boussinesq modeling of wave transformation, breaking, and runup II: 2D. *Journal of Waterway, Port, Coastal, and Ocean Engineering* 2000; **126**(1):48–56.
3. Lin P. Numerical modeling of breaking waves. *Ph.D. Dissertation*, Cornell University, Ithaca, 1998.
4. Zhao Q, Armfield S, Tanimoto K. Numerical simulation of breaking waves by a multi-scale turbulence model. *Coastal Engineering* 2000; **53**(1):53–80.
5. Lynett P, Liu PL-F. A numerical study of submarine-landslide-generated waves and run-up. *Proceedings of the Royal Society of London* 2002; **458**:2885–2910.

DOI: 10.1002/fld

6. Wei G, Kirby JT, Grilli ST, Subramanya R. A fully nonlinear Boussinesq model for surface waves, part 1. Highly nonlinear unsteady waves. *Journal of Fluid Mechanics* 1995; **294**:71–92.
7. Wilson TV. How surfing works. Available from: http://static.howstuffworks.com/gif/surfing-10.gif (accessed on 04/15/08).
8. Nwogu O. Alternative form of Boussinesq equations for nearshore wave propagation. *Journal of Waterway*, *Port*, *Coastal*, *and Ocean Engineering* 1993; **119**(6):618–638.
9. Peregrine DH. Long waves on a beach. *Journal of Fluid Mechanics* 1967; **27**(4):927–935.
10. Press WH, Teukolsky SA, Vetterling WT, Flannery BP. *Numerical Recipes in Fortran 77* (2nd edn). Cambridge University Press: Cambridge, 1992.
11. Hsiao S-C, Lynett P, Hwung H-H, Liu PL-F. Numerical simulations of nonlinear short waves using a multilayer model. *Journal of Engineering Mechanics* 2005; **131**(3):231–243.
12. Rodi W. *Turbulence Models and their Application in Hydraulics*: *A State-of-the-Art Review*. A.A. Balkema: Rotterdam, Netherlands, 1993.
13. Lin P, Liu PL-F. A numerical study of breaking waves in the surfzone. *Journal of Fluid Mechanics* 1998; **359**:239–264.
14. Kothe DB, Mjolness RC, Torrey MD. RIPPLE: a computer program for incompressible flows with free surfaces. *LA-12007-MS*, Los Alamos National Laboratory, 1991.
15. Griebel M, Dornseifer T, Neunhoeffer T. *Numerical Simulation in Fluid Dynamics. A Practical Introduction*. SIAM: Philadelphia, U.S.A., 1998.
16. Dean RG, Dalrymple RA. *Water Wave Mechanics for Engineers and Scientists*. World Scientific Publishing: Singapore, 1991.
17. Wei G, Kirby JT. Alternative form of Boussinesq equations for nearshore wave propagation. *Journal of Waterway*, *Port*, *Coastal*, *and Ocean Engineering* 1995; **121**(5):251–261.
18. Saville TJ. Laboratory data on wave run-up and overtopping on shore structures. *Tech. Memo No. 64*, U.S. Army, Beach Erosion Board, Document Service Center, Dayton, Ohio, 1955.
19. Kobayashi N, Wurjanto A. Wave overtopping on coastal structures. *Journal of Waterway*, *Port*, *Coastal*, *and Ocean Engineering* 1989; **115**(2):235–251.
20. Dodd N. Numerical model of wave run-up, overtopping, and regeneration. *Journal of Waterway*, *Port*, *Coastal*, *and Ocean Engineering* 1999; **124**(2):73–81.

DOI: 10.1002/fld