Chapter 2

2D Tsunami Analysis Model

2.1 Introduction

Tsunamis occur mainly by the significant displacement of sea floor due to earthquakes or volcanic eruptions. By the distance of propagation, tsunamis can be classified into near-field tsunami and trans-ocean tsunami [37]-[40].

For a near-field tsunami in deep sea until it arrives at a water depth of about 50m [40], the linear long-wave theory gives satisfactory results. When the water depth is shallower than 50m, the nonlinear effects become important, and the nonlinear long-wave theory including the bottom friction term is needed. However, due to the nonlinearity, the front of wave becomes steepen and breaking. And as the approximation of the horizontal velocity is constant throughout the depth of the sea, the breaking cannot be reproduced by the nonlinear long-wave theory. Therefore, in practical numerical simulation, the Gibbs oscillations may be occurred [37]. In this case, the nonlinear dispersion wave theory sometimes can help to improve this issue, because the dispersibility can draw back the steep wave crest. In addition, because the nonlinear dispersion wave theory also cannot reproduce the wave breaking, when the wave height is amplified into breaking wave due to the soliton fission, the nonlinear dispersion wave theory cannot be satisfactory, a special breaking wave model (for example, [41],[42]) is needed to be introduced.

A trans-ocean tsunami is a tsunami that travels from the source more than 1,000km and 3 hours tsunami travel time [43], such as 2004 Indian Ocean tsunami.

For this kind of tsunami, the propagation in deep ocean, the linear dispersion wave theory with the Coriolis force, expressed in the latitude-longitude coordinate system, is satisfactorily used [39],[44],[45].

Timely prediction of tsunami runup for issuing tsunami warnings is the best way to save human lives, provided the tsunami simulation is a powerful tool to make prediction. To determine the wave arrival time, the coastal wave height and the inundation area for tsunamis, simulations of the 2D shallow water equations can be useful because they take short computational time and get satisfactory results. Numerous numerical methods have been proposed to solve the equations based on the wave theories mentioned above and the effectiveness has been shown, such as the finite difference method based on staggered Leap-Frog scheme [46],[47], the finite element method [48]-[50].

In this chapter, the general procedure of the stabilized finite element method [48] is introduced to solve the 2D linear/nonlinear shallow water equations and linear/nonlinear Boussinesq equations. For the numerical examples, the propagation of solitary wave problem and the wave-making problem are simulated for the model verification and validation. Finally, the applicability of the present 2D models are demonstrated by simulating the large-scale tsunami runup due to the 2011 Great East Japan Earthquake.
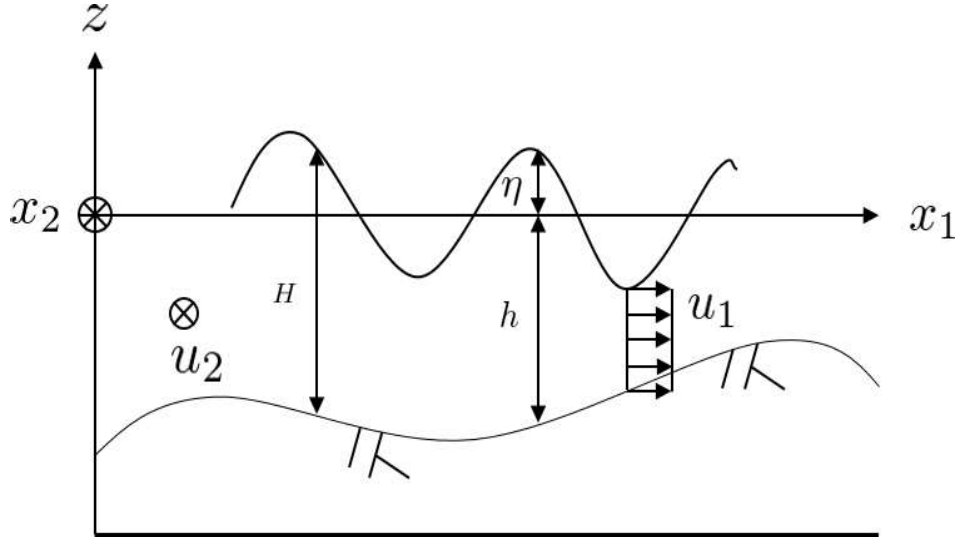
2.2 Governing Equations

Figure **2.1** **Coordinate system**

In order to investigate the characteristic of the wave forms during wave propagation, the governing equations based on the linear long-wave theory, the nonlinear long-wave theory, the linear dispersion wave theory, the nonlinear dispersion wave theory [51] are employed. The coordinate system is shown in **Figure 2.1**.

2.2.1 Linear Shallow Water Equations

The linear shallow water equations (L-SWE) based on the linear long-wave theory are given by,

$$\frac{\partial \mathbf{U}}{\partial t} + \mathbf{A}_i \frac{\partial \mathbf{U}}{\partial x_i} = \mathbf{R}, \tag{2.1}$$

where the matrix terms can be expressed as,

$$\mathbf{U} = \begin{bmatrix} H \\ u_1 h \\ u_2 h \end{bmatrix}, \mathbf{R} = \begin{bmatrix} 0 \\ -c^2 \frac{\partial z}{\partial x_1} \\ -c^2 \frac{\partial z}{\partial x_2} \end{bmatrix},$$

$$\mathbf{A}_1 = \begin{bmatrix} 0 & 1 & 0 \\ c^2 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \mathbf{A}_2 = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ c^2 & 0 & 0 \end{bmatrix},$$

where u_i is the average velocity in $x_i (i = 1, 2)$ directions. H is the total water depth, h is the static water depth, c is the wave speed $(= \sqrt{gh})$, g is the gravitational

acceleration and z is the height of bottom. $\mathbf{U}$, $\mathbf{R}$, and $\mathbf{A_i}$ are the unknown matrix, source term vector, and advection matrix, respectively.

2.2.2   Nonlinear Shallow Water Equations

The nonlinear shallow water equations (N-SWE) based on the nonlinear long-wave theory are given by,

$$\frac{\partial \mathbf{U}}{\partial t} + \mathbf{A}_i \frac{\partial \mathbf{U}}{\partial \mathbf{x}_i} - \frac{\partial}{\partial \mathbf{x}_i}\left(\mathbf{N}_{ij}\frac{\partial \mathbf{U}}{\partial \mathbf{x}_j}\right) + \mathbf{G}\mathbf{U} = \mathbf{R}, \qquad (2.2)$$

where the matrix terms can be expressed as,

$$\mathbf{U} = \begin{bmatrix} H \\ u_1 H \\ u_2 H \end{bmatrix}, \mathbf{R} = \begin{bmatrix} 0 \\ -c^2\frac{\partial z}{\partial x_1} \\ -c^2\frac{\partial z}{\partial x_2} \end{bmatrix},$$

$$\mathbf{A}_1 = \begin{bmatrix} 0 & 1 & 0 \\ c^2 - u_1^2 & 2u_1 & 0 \\ -u_1 u_2 & u_2 & u_1 \end{bmatrix}, \mathbf{A}_2 = \begin{bmatrix} 0 & 0 & 1 \\ -u_1 u_2 & u_2 & u_1 \\ c^2 - u_2^2 & 0 & 2u_2 \end{bmatrix},$$

$$\mathbf{N}_{11} = \nu_e \begin{bmatrix} 0 & 0 & 0 \\ -2u_1 & 2 & 0 \\ -u_2 & 0 & u_1 \end{bmatrix}, \mathbf{N}_{12} = \nu_e \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ -u_1 & 1 & 0 \end{bmatrix},$$

$$\mathbf{N}_{21} = \nu_e \begin{bmatrix} 0 & 0 & 0 \\ -u_2 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}, \mathbf{N}_{22} = \nu_e \begin{bmatrix} 0 & 0 & 0 \\ -u_1 & 1 & 0 \\ -2u_2 & 0 & 2 \end{bmatrix},$$

$$\mathbf{G} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & \frac{C_f\sqrt{u_1^2 + u_2^2}}{H} & 0 \\ 0 & 0 & \frac{C_f\sqrt{u_1^2 + u_2^2}}{H} \end{bmatrix},$$

$$C_f = \frac{gn^2}{H^{\frac{1}{3}}}, \qquad H = h + \eta, \qquad c = \sqrt{gH},$$

where ν_e is the eddy viscosity coefficient, n is the Manning's roughness coefficient and η is the water level variation. $\mathbf{N_{ij}}$, $\mathbf{G}$ are the diffusion and friction matrices, respectively.

2.2.3   Linear Boussinesq Equations

The linear Boussinesq equations (L-BE) based on the linear dispersion wave theory are given by,

$$\frac{\partial \mathbf{U}}{\partial t} + \mathbf{A}_i \frac{\partial \mathbf{U}}{\partial x_i} = \frac{\partial^2}{\partial t \partial^2 x_i}(\mathbf{K}) + \mathbf{R}, \tag{2.3}$$

where,

$$\mathbf{U} = \begin{bmatrix} H \\ u_1 h \\ u_2 h \end{bmatrix}, \mathbf{R} = \begin{bmatrix} 0 \\ -c^2 \frac{\partial z}{\partial x_1} \\ -c^2 \frac{\partial z}{\partial x_2} \end{bmatrix}, \mathbf{K} = \begin{bmatrix} 0 \\ \frac{h^2}{3} \frac{\partial u_i h}{\partial x_1} \\ \frac{h^2}{3} \frac{\partial u_i h}{\partial x_2} \end{bmatrix},$$

$$\mathbf{A}_1 = \begin{bmatrix} 0 & 1 & 0 \\ c^2 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix}, \mathbf{A}_2 = \begin{bmatrix} 0 & 0 & 1 \\ 0 & 0 & 0 \\ c^2 & 0 & 0 \end{bmatrix},$$

where $\mathbf{K}$ is the dispersion matrix.

2.2.4   Nonlinear Boussinesq Equations

The nonlinear Boussinesq equations (N-BE) based on the nonlinear dispersion wave theory are given by,

$$\frac{\partial \mathbf{U}}{\partial t} + \mathbf{A}_i \frac{\partial \mathbf{U}}{\partial \mathbf{x}_i} - \frac{\partial}{\partial \mathbf{x}_i}\left(\mathbf{N}_{ij}\frac{\partial \mathbf{U}}{\partial \mathbf{x}_j}\right) + \mathbf{GU} = \frac{\partial^2}{\partial t \partial \mathbf{x}_i}(\mathbf{K}) + \mathbf{R}, \tag{2.4}$$

where the matrix terms can be expressed as:

$$\mathbf{U} = \begin{bmatrix} H \\ u_1 H \\ u_2 H \end{bmatrix}, \mathbf{K} = \begin{bmatrix} 0 \\ \frac{h^2}{3} \frac{\partial u_i H}{\partial x_1} \\ \frac{h^2}{3} \frac{\partial u_i H}{\partial x_2} \end{bmatrix}, \mathbf{R} = \begin{bmatrix} 0 \\ -c^2 \frac{\partial z}{\partial x_1} \\ -c^2 \frac{\partial z}{\partial x_2} \end{bmatrix},$$

$$\mathbf{A}_1 = \begin{bmatrix} 0 & 1 & 0 \\ c^2 - u_1^2 & 2u_1 & 0 \\ -u_1 u_2 & u_2 & u_1 \end{bmatrix}, \mathbf{A}_2 = \begin{bmatrix} 0 & 0 & 1 \\ -u_1 u_2 & u_2 & u_1 \\ c^2 - u_2^2 & 0 & 2u_2 \end{bmatrix},$$

$$\mathbf{N}_{11} = \nu_e \begin{bmatrix} 0 & 0 & 0 \\ -2u_1 & 2 & 0 \\ -u_2 & 0 & u_1 \end{bmatrix}, \mathbf{N}_{12} = \nu_e \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ -u_1 & 1 & 0 \end{bmatrix},$$

$$\mathbf{N}_{21} = \nu_e \begin{bmatrix} 0 & 0 & 0 \\ -u_2 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}, \mathbf{N}_{22} = \nu_e \begin{bmatrix} 0 & 0 & 0 \\ -u_1 & 1 & 0 \\ -2u_2 & 0 & 2 \end{bmatrix},$$

$$\mathbf{G} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & \frac{C_f\sqrt{u_1^2+u_2^2}}{H} & 0 \\ 0 & 0 & \frac{C_f\sqrt{u_1^2+u_2^2}}{H} \end{bmatrix}, C_f = \frac{gn^2}{H^{\frac{1}{3}}}.$$

The parameters can be refered to the former sections.

2.3   Stabilized Finite Element Method

2.3.1   SUPG (Streamline Upwind/Petrov-Galerkin) Method

The shallow water equations and the Boussinesq equations are kinds of advection-diffusion equations. In the case of discretizing the governing equations by the general Galerkin finite element method, oscillation occurs when advection is dominant. In order to resolve this issue, an Upwind Galerkin method has been proposed by adding artificial diffusion to weight function in the flow direction for the finite element discretization. Furthermore, a stabilized finite element method has been developed based on the Streamline-Upwind/Petrov-Galerkin (SUPG) method [52],[53] which adds an appropriate amount of artificial diffusion only into the streamline direction of the flow.

In the SUPG method, the weight function $\widetilde{\omega}_i$ for the momentum equations is given by:

$$\widetilde{\omega}_i^h = \omega_i^h + \delta_i^h, \qquad \delta_i^h = \tau_m u_j^h \frac{\partial \omega_i^h}{\partial x_j}, \tag{2.5}$$

where τ_m is a stabilization parameter with time dimension. **Figures 2.2** and **2.3** show the case of one dimension primary element. By comparing to the general weight function, the weight function is larger by δ at the upwind side, and it is smaller by δ
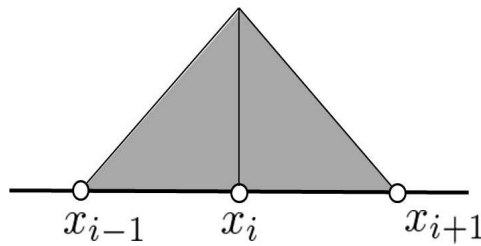

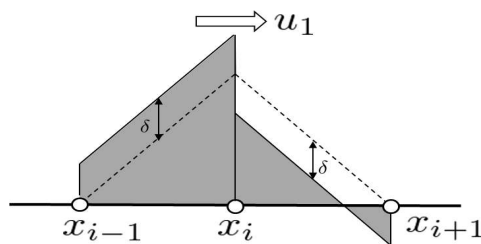Figure **2.2**   **Weight function for Galerkin method**

Figure **2.3**   **Weight function for SUPG method**

at the downwind side.

2.3.2   Shock-Capturing Term

The computation of the flows with shock waves is a difficult task because such kind of flows are resulting in sharp and discontinuous changes in water surface. In order to resolve the issues of overshoot and undershoot occurring at the discontinuous surface, shock-capturing is needed. The shock-capturing term [54]-[56] is 'a certain artificial viscosity with viscosity coefficient depending (locally) on the residual of the finite element solution, where the residual is obtained by inserting the finite element solution into the given hyperbolic differential equation' [57]. The main idea of the shock-capturing is to add significant artificial viscosity to a discontinuity where the residual is large, but little in smooth regions where the residual is small. The typical format of the shock-capturing term can be written as Eq. **(2.6)** [1],

$$\sum_{e=1}^{n_{el}} \delta \int_{\Omega_e} \frac{\partial \Phi^*}{\partial x} \frac{\partial \Phi}{\partial x} d\Omega, \tag{2.6}$$

where Φ is a physical quantity and Φ^* is the weight function. δ is a shock-capturing parameter which can be defined as [1],[58]:

$$\delta = \frac{||\Phi||h_e}{2} \xi(\mathrm{Re}_e), \tag{2.7}$$

$$\mathrm{Re}_e = \frac{||\mathbf{u}||h_e}{2\nu}, \tag{2.8}$$

$$\xi(\mathrm{Re}_e) = \left\{ \begin{array}{ll} \frac{\mathrm{Re}_e}{3}, & \mathrm{Re}_e \leq 3, \\ 1, & \mathrm{Re}_e > 3. \end{array} \right. \tag{2.9}$$

where h_e is a measure of element length, Re_e is the Reynold number of an element, and ν is the dynamic viscosity.

Figure 2.4 shows the effectiveness of the stabilized terms (i.e., the SUPG and shock-capturing terms).
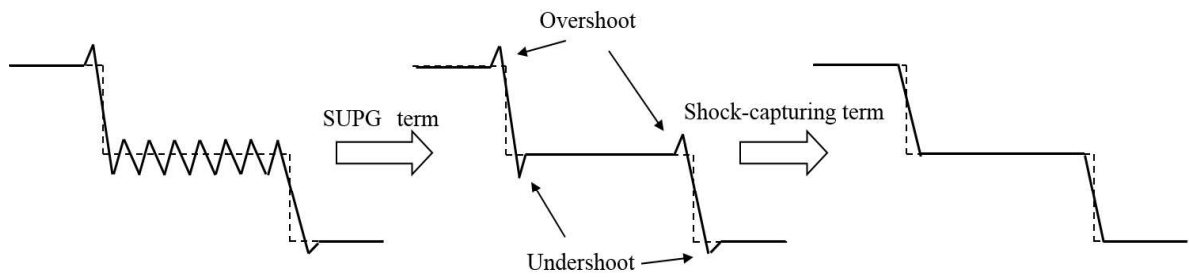
Figure **2.4**   **Effectiveness of stabilized terms** [56]

Oscillations can be observed by using the Galerkin method (left), and can be reduced by the SUPG term (center). Furthermore, the remaining overshoot and undershoot at the discontinuous surface can be prevented by the shock-capturing term.

2.4   Spatial Discretization

To apply the stabilized finite element method based on the SUPG method to the governing Eq. **(2.4)**, the following weighted residual equations can be obtained,

$$
\int_\Omega \mathbf{U}^* \cdot \Big(\frac{\partial \mathbf{U}}{\partial t} + \mathbf{A}_i \frac{\partial \mathbf{U}}{\partial \mathbf{x}_i} - \mathbf{R}\Big) d\Omega + \int_\Omega \Big(\frac{\partial \mathbf{U}^*}{\partial \mathbf{x}_i}\Big) \cdot \Big(\mathbf{N}_{ij} \frac{\partial \mathbf{U}}{\partial \mathbf{x}_j} + \frac{\partial}{\partial t}(\mathbf{K})\Big) d\Omega
$$
$$
+ \sum_{e=1}^{n_{el}} \int_{\Omega_e} \tau(\mathbf{A}_j)^T \Big(\frac{\partial \mathbf{U}^*}{\partial \mathbf{x}_j}\Big) \cdot \Big(\frac{\partial \mathbf{U}}{\partial t} + \mathbf{A}_i \frac{\partial \mathbf{U}}{\partial \mathbf{x}_i} + \mathbf{GU} - \mathbf{R}\Big) d\Omega
$$
$$
+ \sum_{e=1}^{n_{el}} \int_{\Omega_e} \delta\Big(\frac{\partial \mathbf{U}^*}{\partial \mathbf{x}_i}\Big) \cdot \Big(\frac{\partial \mathbf{U}}{\partial \mathbf{x}_i}\Big) d\Omega = \int_\Gamma \mathbf{U}^* \mathbf{T} d\Gamma, \tag{2.10}
$$

where the first and the second terms are the Galerkin terms, the third term is the stabilized term based on the SUPG method and the fourth term is the shock capturing term. τ, δ are the stabilized parameters which can be written as follow [48],[59],

$$
\tau = \Big(\frac{1}{(\tau_{\mathrm{SUGN1}})^2} + \frac{1}{(\tau_{\mathrm{SUGN2}})^2}\Big)^{-\frac{1}{2}},
$$
$$
\delta = \tau_{\mathrm{SHOC}} \left(||\mathbf{u}_{\mathrm{int}}||\right)^2,
$$

where,

$$
\tau_{\mathrm{SUGN1}} = \Big(\sum_{a=1}^{n_{el}}(c\,|\,\mathbf{j}\cdot\nabla N_a\,| + |\,\mathbf{u}\cdot\nabla N_a\,|)\Big)^{-1},
$$
$$
\tau_{\mathrm{SUGN2}} = \frac{\Delta t}{2},
$$
$$
\mathbf{j} = \frac{\nabla H}{\|\,\nabla H\,\|},
$$
$$
\tau_{\mathrm{SHOC}} = \Big(\sum_{\alpha=1}^{n_{en}}\Big|\mathbf{u}\cdot\nabla N_\alpha\Big|\Big)^{-1},
$$
$$
||\mathbf{u}_{\mathrm{int}}|| = ||\mathbf{u}||,
$$

where Δt is the time increment, N_a is the shape function, and c is the wave speed.

Using the first order triangle element (see **Figure 2.5**) to make interpolation for Eq. **(2.10)**, the following finite element equations can be obtained.

$$
(\mathbf{M} + \mathbf{M}_\tau + \mathbf{D})\dot{\mathbf{U}} + (\mathbf{K} + \mathbf{K}_\tau)\mathbf{U} + \mathbf{SU}
$$
$$
+ (\mathbf{F} + \mathbf{F}_\tau)\mathbf{U} + (\mathbf{H} + \mathbf{H}_\tau)\mathbf{R} + \mathbf{TU} = 0, \tag{2.11}
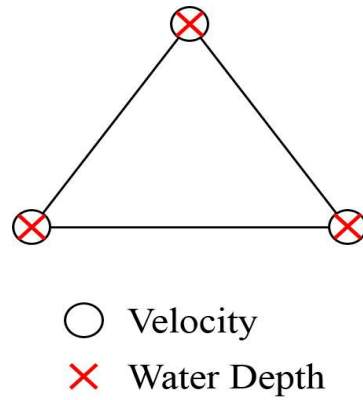$$

Figure **2**.5   **First order triangle element (P1/P1 element)**

where $\mathbf{M}$, $\mathbf{D}$, $\mathbf{K}$, $\mathbf{S}$, $\mathbf{F}$, $\mathbf{H}$, $\mathbf{T}$ are mass matrix, dispersion matrix, advection matrix, diffusion matrix, friction matrix, gradient matrix, shock capturing matrix. Matrices with subscript indicate them caused by the SUPG method.

2.5   Temporal Discretization

A finite difference method is applied for the temporal discretization. For the time step of $n + \theta$,

$$\dot{\mathbf{U}}^{n+\theta} = \frac{1}{\Delta t}(\mathbf{U}^{n+1} - \mathbf{U}^n), \tag{2.12}$$

$$\mathbf{U}^{n+\theta} = (1 - \theta)\mathbf{U}^n + \theta\mathbf{U}^{n+1}. \tag{2.13}$$

Depending on the θ, the temporal discretization is divided into three cases,

- $\theta = 0$ explicit method, forward difference (first order accuracy)
- $\theta = 0.5$ implicit method, Crank-Nicolson method (second order accuracy)
- $\theta = 1.0$ implicit method, backward difference (first order accuracy)

In this study, the Crank-Nicolson method is applied to the equation (**2.11**), the following equations can be obtained,

$$(\mathbf{M} + \mathbf{M}_\tau + \mathbf{D})\mathbf{U}^{n+1} + \frac{\Delta t}{2}(\mathbf{K} + \mathbf{K}_\tau + \mathbf{F} + \mathbf{F}_\tau + \mathbf{S} + \mathbf{T})\mathbf{U}^{n+1}$$

$$= (\mathbf{M} + \mathbf{M}_\tau + \mathbf{D})\mathbf{U}^n - \frac{\Delta t}{2}(\mathbf{K} + \mathbf{K}_\tau + \mathbf{F} + \mathbf{F}_\tau + \mathbf{S} + \mathbf{T})\mathbf{U}^n - \Delta t(\mathbf{H} + \mathbf{H}_\tau)\mathbf{R}. \tag{2.14}$$

To solve the simultaneous linear equations (**2.14**), the diagonal scaling preprocessing method [60], the element-by-element processing [61], and the Bi-CGSTAB (Bi-Conjugate Gradient STABilized) method [62],[63] are applied.

2.6   Wetting and Drying Treatment

For the simulation of flood in river or tsunami, it is necessary to consider the behavior of the shoreline. In this study, the Eulerian method [56],[64] which is excellent in applicability to complex topography was applied. The algorithm is shown below.

Firstly, the area assumed to be flooded is divided into finite elements. Secondly, comparing the water depth H^n of all nodes with a critical water depth ε by every time step to decide whether the elements should be computed. The comparison is described as follow (refer to **Figure 2.6**):
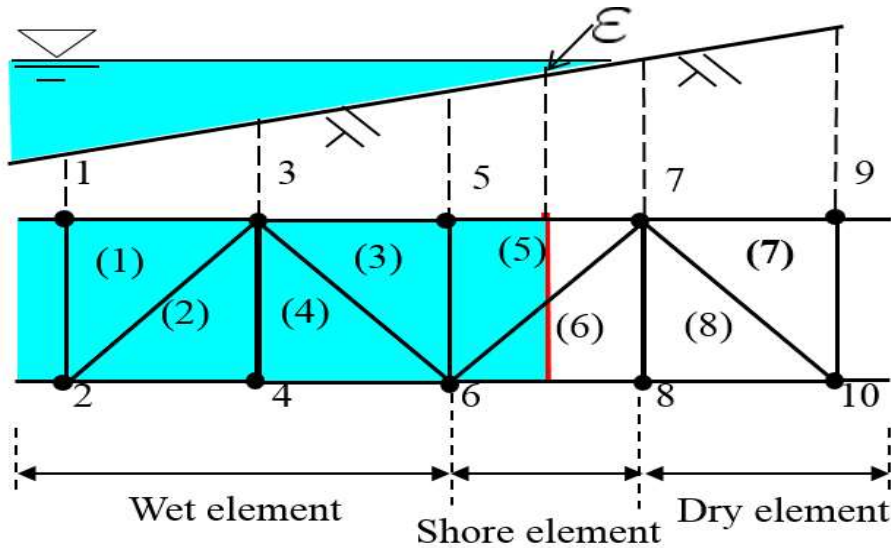
Figure 2.6   **Judgement for wet or dry element**

- If $H < \varepsilon$ for all the nodes of an element, the element is denoted as a dry element (such as the elements (7) and (8) in Figure 2.6) which is not computed.
- If $H \geqq \varepsilon$ for all the nodes of an element, the element is denoted as a wet element (such as the elements (1), (2), (3), (4) in Figure 2.6) which is computed.
- If $H \geqq \varepsilon$ for one or two nodes of an element, the element is denoted as a shore element (such as the elements (5) and (6) in Figure 2.6) which is computed. And for the nodes where $H < \varepsilon$, the water depths of the nodes are set to be ε, and the velocities of the nodes are set to be 0.

2.7 Numerical Examples

2.7.1 Propagation of Solitary Wave
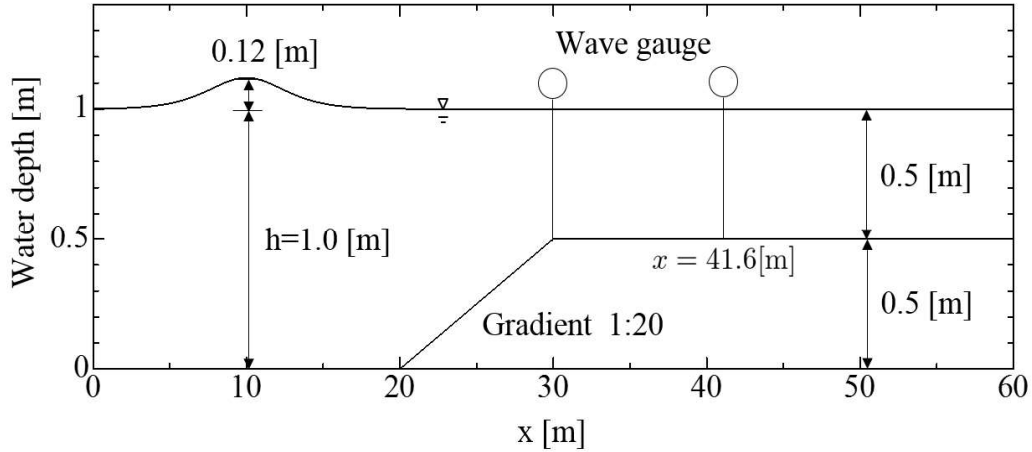
Figure 2.7   **Computational model**

In order to investigate the characteristic of the wave forms during propagation and the effectiveness of the terms of the governing equations, the propagation of solitary wave [65] is simulated (see **Figure 2.7**). For the physical constants, we set the gravitational acceleration $g = 9.8\text{m/s}^2$ and the Manning's roughness coefficient n=0.01s/m$^{\frac{1}{3}}$. For the computational condition, the mesh size is 0.1m and the time increment is 0.005s.

A comparison of the numerical results are shown in **Figure 2.8**. From the figure, we can see: (1) the wave hight increase because of the wave shoaling but the wave is symmetry for the L-SWE case. (2) the wave speed is the fastest because of the wave bending forward in the N-SWE case. (3) dispersion waves can be observed in the L-BE case. (4) For the N-BE case, the wave bending forward because of the nonlinearity but it is less remarkable than the N-SWE. The wave is higher than the N-SWE. **Figure 2.9**, **Figure 2.10** show the comparison of the computational results with the experimental results [65] at $x/h = 30.0$ and $x/h = 41.6$. We can see the results of N-BE show the best agreement with the experimental results in all the simulation results.
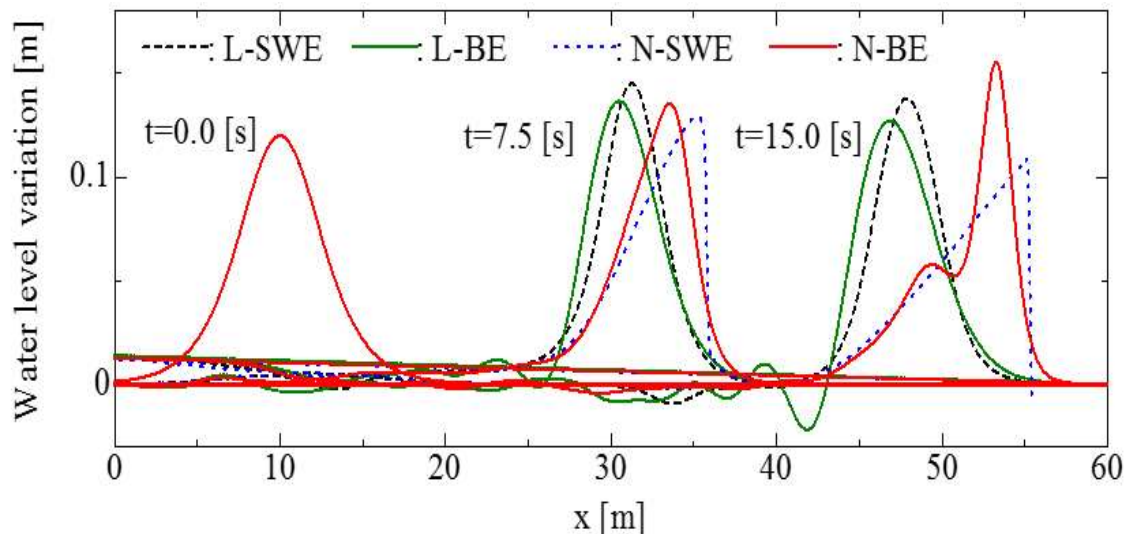
Figure **2.8** **Comparison of the results by different equations**
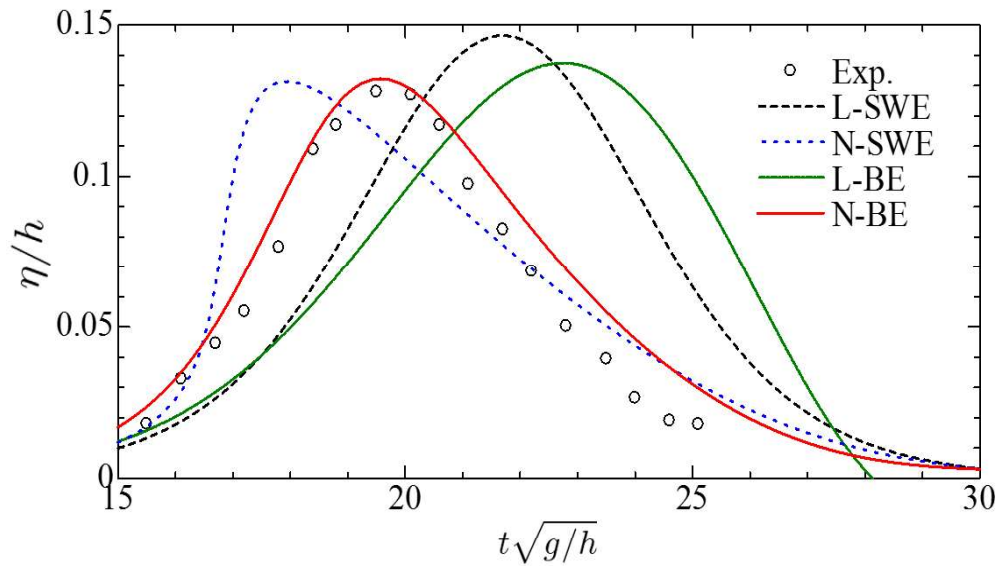
Figure 2.9 **Comparison of time series results at** $x/h = 30.0$
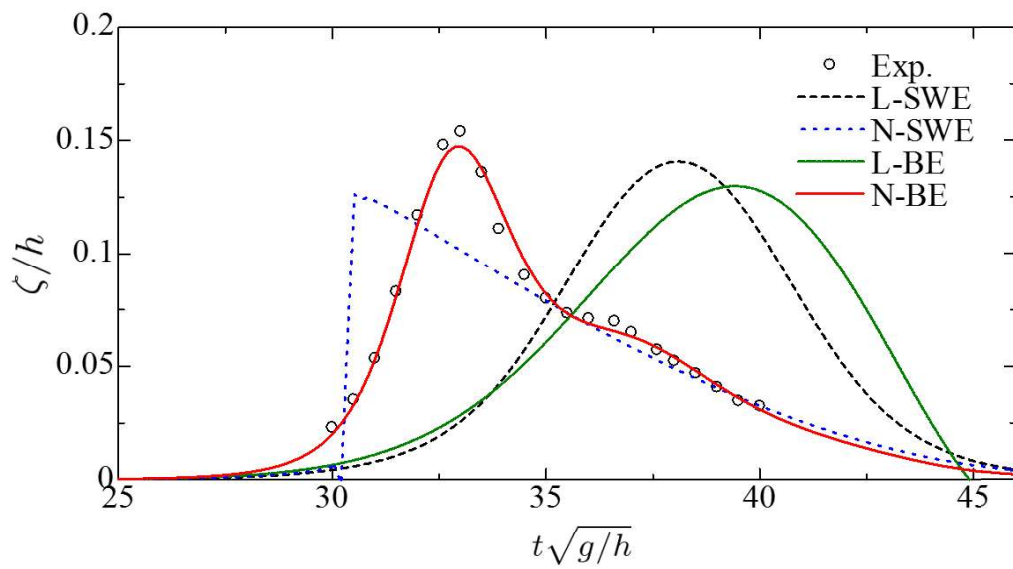
Figure **2.10**   **Comparison of time series results at** $x/h = 41.6$

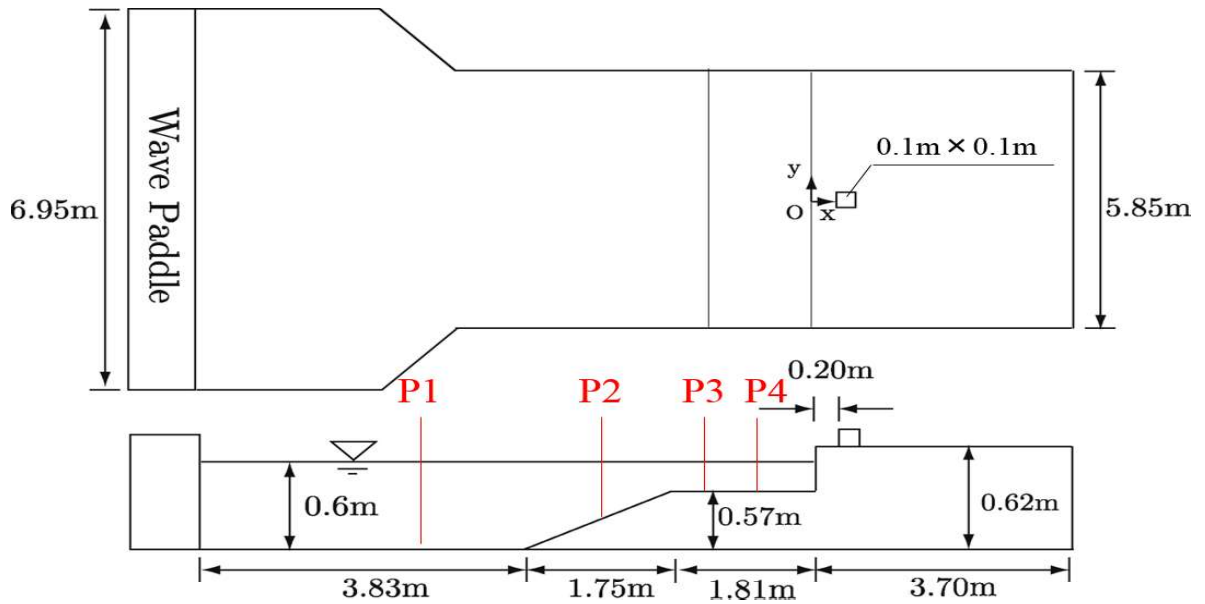## 2.7.2   Wave-Making Problem

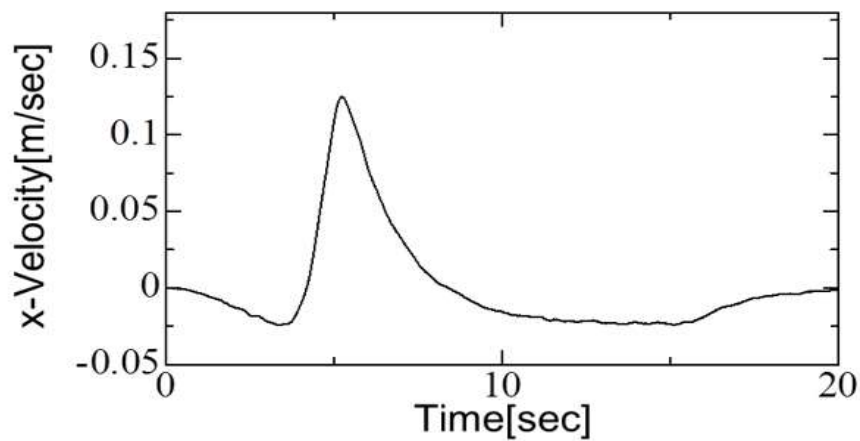Figure 2.11   **Wave-making problem**

Figure 2.12   **Velocity of the wave paddle**

In order to evaluate effectiveness and applicability of the 2D N-SWE and N-BE models, a wave-making problem [66] is simulated. The computational model is shown in **Figure 2.11**. In this model, a wave paddle is located at the left side to make waves.

Figure 2.12 shows the velocity of the wave paddle. For the analysis conditions, unstructured mesh (nodes: 44,991, elements: 89,313, see **Figure 2.13**) is used. The time increment is 0.001s and the critical water depth is 0.005m. The Manning's roughness coefficient is set to be $0.01 \mathrm{s/m}^{\frac{1}{3}}$. Water elevations are observed at P1 ($x = -3.80$m), P2 ($x = -2.30$m), P3 ($x = -1.25$m), and P4 ($x = -0.75$m).

Comparisons for the time history of the water elevations between the simulated results and the experimental results are shown in **Figure 2.14** and **Figure 2.15**. From the **Figure 2.14**, we can see the peak value of N-BE agree with the experimental results better than that of the N-SWE, but the results are almost the same in the **Figure 2.15**. And both the N-BE and the N-SWE can simulate the deformation of the waves well comparing to the experimental results.
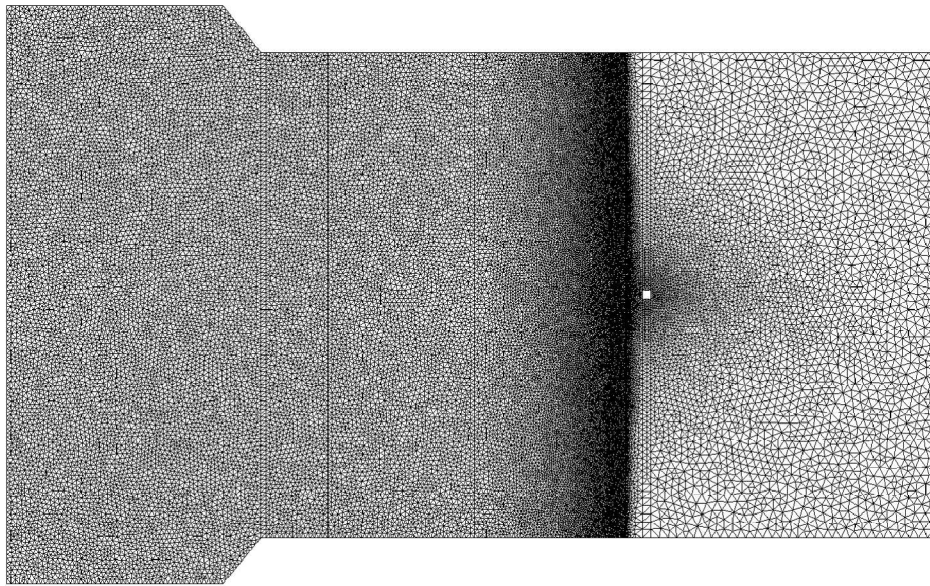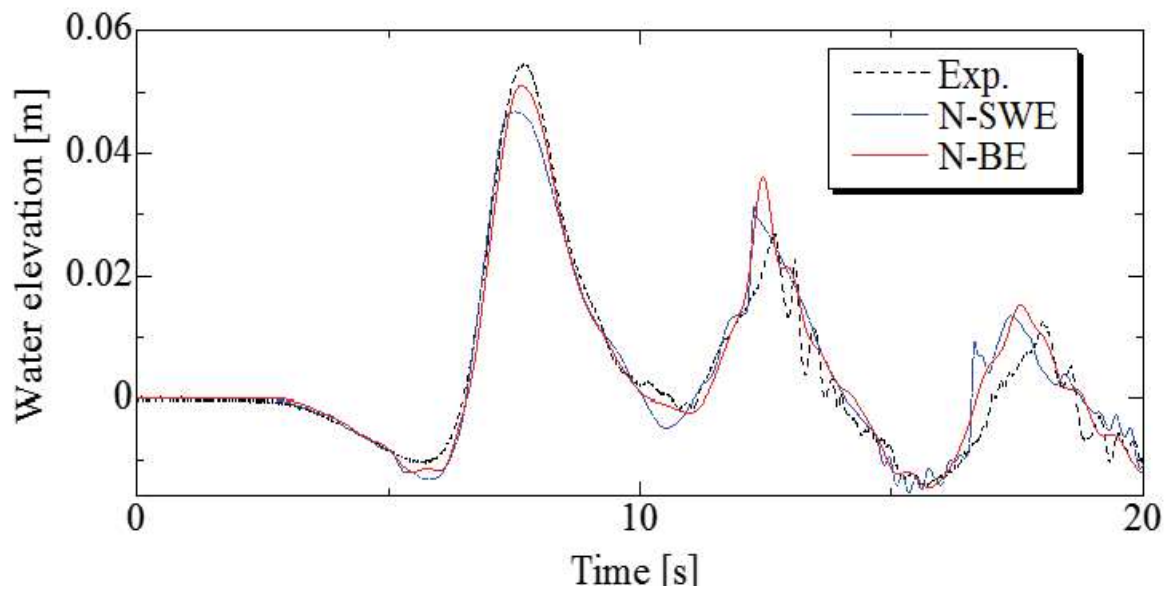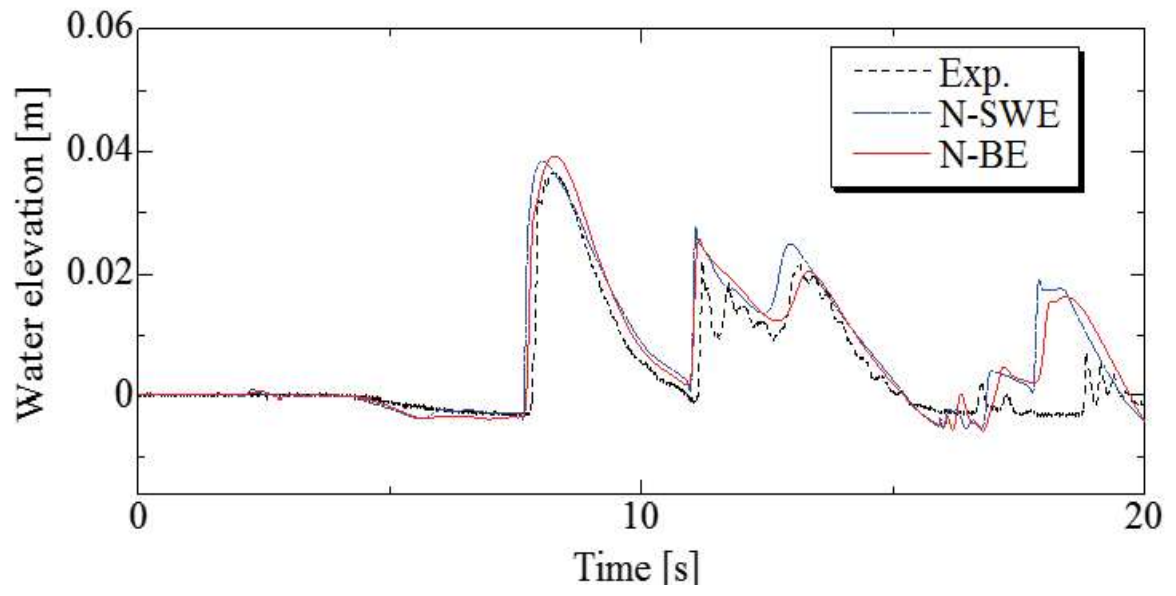

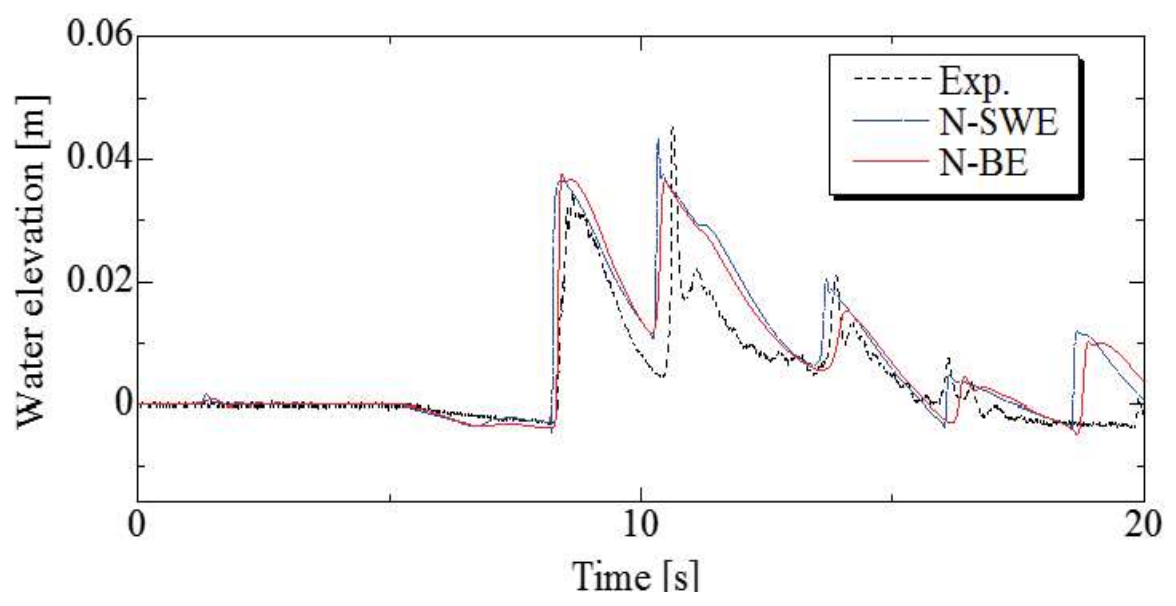Figure 2.13   **Computational mesh**
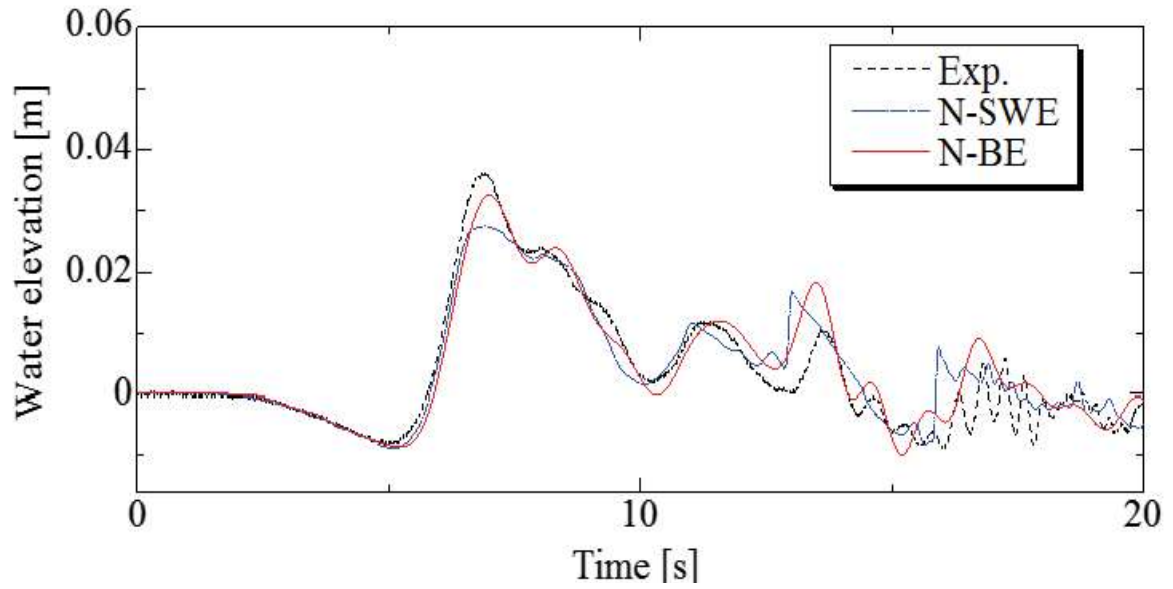
P1 ($x = -3.8$m)

P2 ($x = -2.3$m)

Figure 2.14   Results at P1 and P2

P3 ($x = -1.25$m)

P4 ($x = -0.75$m)

Figure **2**.15 **Results at P3 and P4**

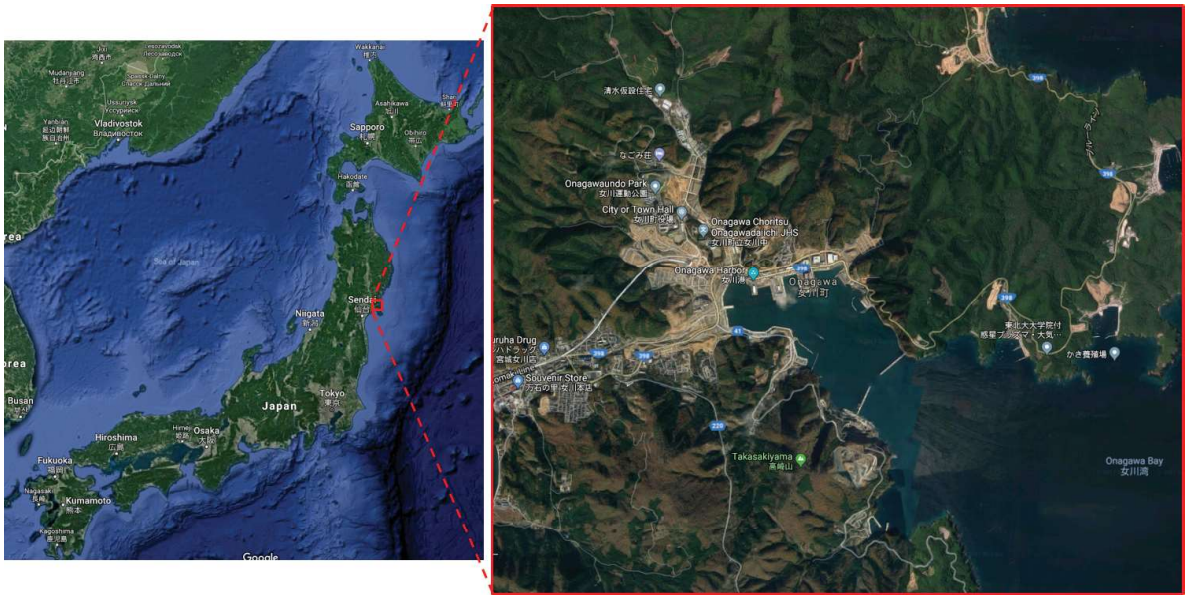## 2.7.3 Large-Scale Tsunami Runup Simulation by a 2D Model

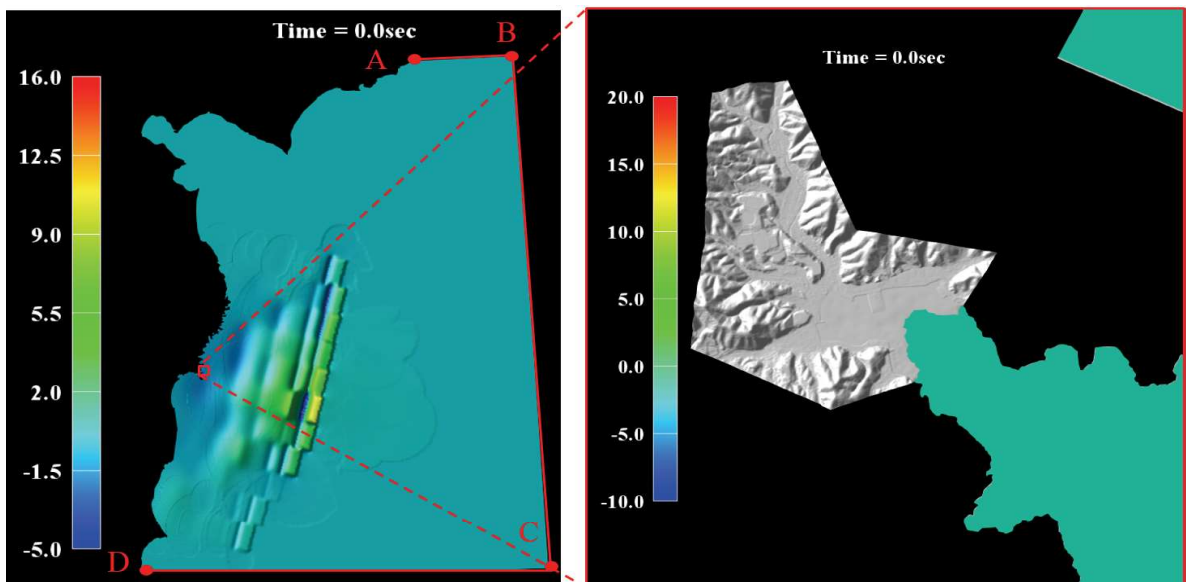Figure 2.16 **Target area for tsunami runup simulation (from GoogleEarth)**

Figure 2.17 **Computational domain and initial condition**

To investigate applicability to tsunami simulation over real terrain by the 2D N-SWE

and BE models, the tsunami inundation in Onagawa District (see **Figure 2.16**) due
to the 2011 Great East Japan Earthquake is simulated. The computational domain
and the initial condition are shown in **Figure 2.17**. The initial condition is computed
by the Mansinha and Smylie method [67], and the fault model of Ver. 8.0 by Satake
et al. [68] is used. For the boundary conditions, the shoreline is non-slip condition,
and the ABCD boundary is an open boundary condition. In this study, the open
boundary condition [69] is governed by the following equations using the water depth
H and the water level variation η,

- Boundary A-B:

$$u_1 = 0, \tag{2.15}$$

$$u_2 = \eta \sqrt{g/H}, \tag{2.16}$$

- Corner node B:

$$u_1 = \eta \sqrt{g/H} \times \frac{1}{\sqrt{2}}, \tag{2.17}$$

$$u_2 = \eta \sqrt{g/H} \times \frac{1}{\sqrt{2}}, \tag{2.18}$$

- Boundary B-C:

$$u_1 = \eta \sqrt{g/H}, \tag{2.19}$$

$$u_2 = 0, \tag{2.20}$$

- Corner node C:

$$u_1 = -\eta \sqrt{g/H} \times \frac{1}{\sqrt{2}}, \tag{2.21}$$

$$u_2 = -\eta \sqrt{g/H} \times \frac{1}{\sqrt{2}}, \tag{2.22}$$

- Boundary C-D:

$$u_1 = 0, \tag{2.23}$$

$$u_2 = -\eta \sqrt{g/H}. \tag{2.24}$$

The computational mesh is generated by the modified Delaunay method [70] based
on a constant Courant number for each element [69],[71], as the result, the width of
element is large in the deep water region and it becomes smaller as the water depth

becomes shallow. The mesh size in the land is about 8m, and the number of nodes is 557,252 and the element is 1,103,938. The time increment is 0.1s. The Manning's roughness coefficient is set to be $0.025 \text{s/m}^{\frac{1}{3}}$.

The analysis results of wave propagation at t=700s, 1400s, 2700s, 3600s are shown in **Figure 2.18**, **Figure 2.19**. We can see the waves dispersing obviously in the results by using N-BE than the N-SWE. **Figure 2.20** shows the inundation areas of the computational results at t=2500s when the highest waves reach the hill. We can see the runup height is quite similar to the observed data shown in **Figure 2.21** [1]. The largest inundation areas are shown in **Figure 2.22** by the computations. We can see the inundation areas by using the N-SWE are almost the same as that by the N-BE. And the inundation areas are in agreement with the observation data shown in **Figure 2.23**[1]. However, some differences can be found in the front of the inundation area. This is probably due to the mesh size or the buildings. To investigate about the differences by using a finer mesh or computing the urban area in 3D can be a future work.

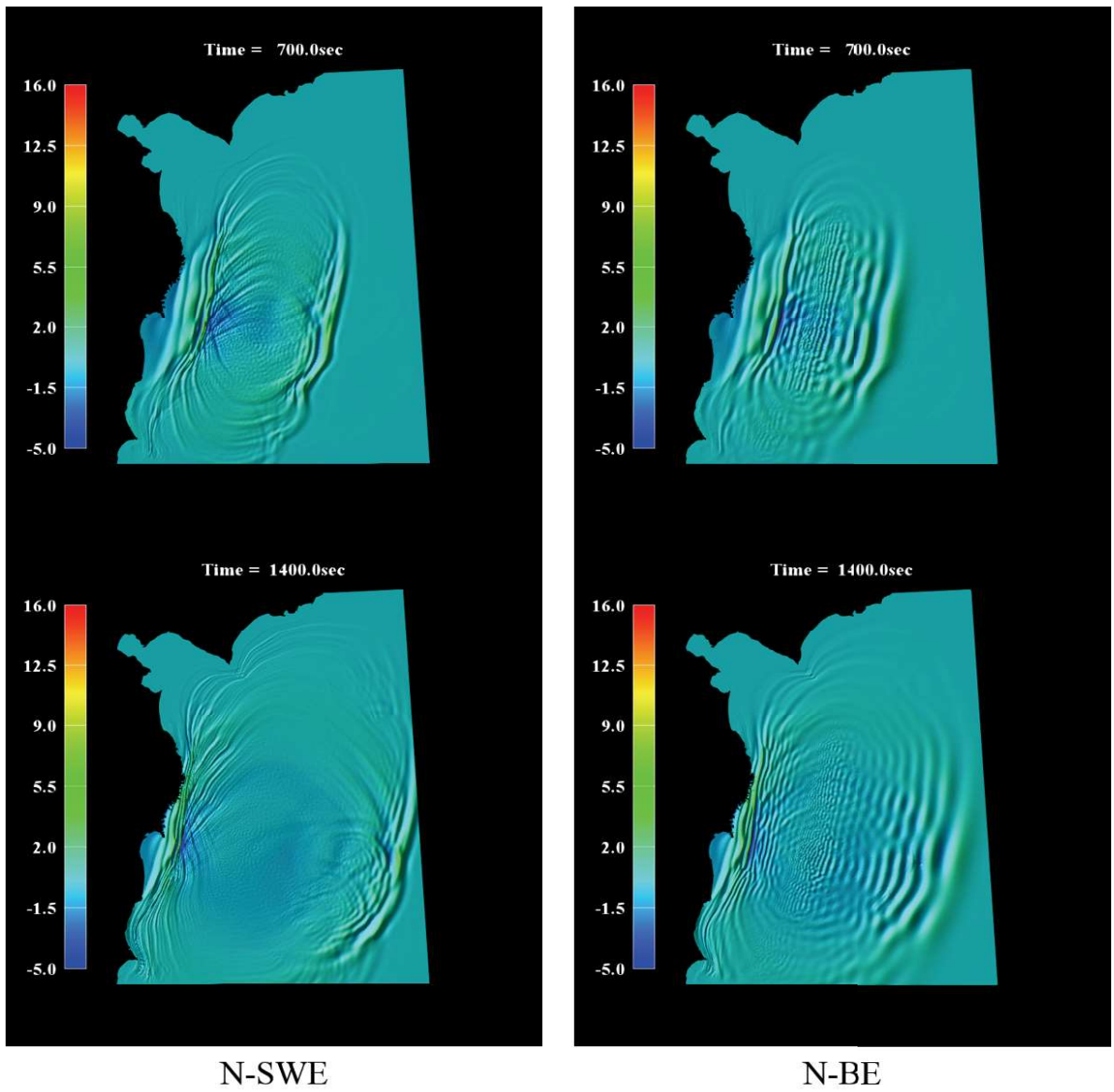[1] `http://133.6.118.74/map/map/?mid=10&cid=2&gid=0&lon=141.97198&lat=39.73765&scale=150000`

N-SWE   N-BE
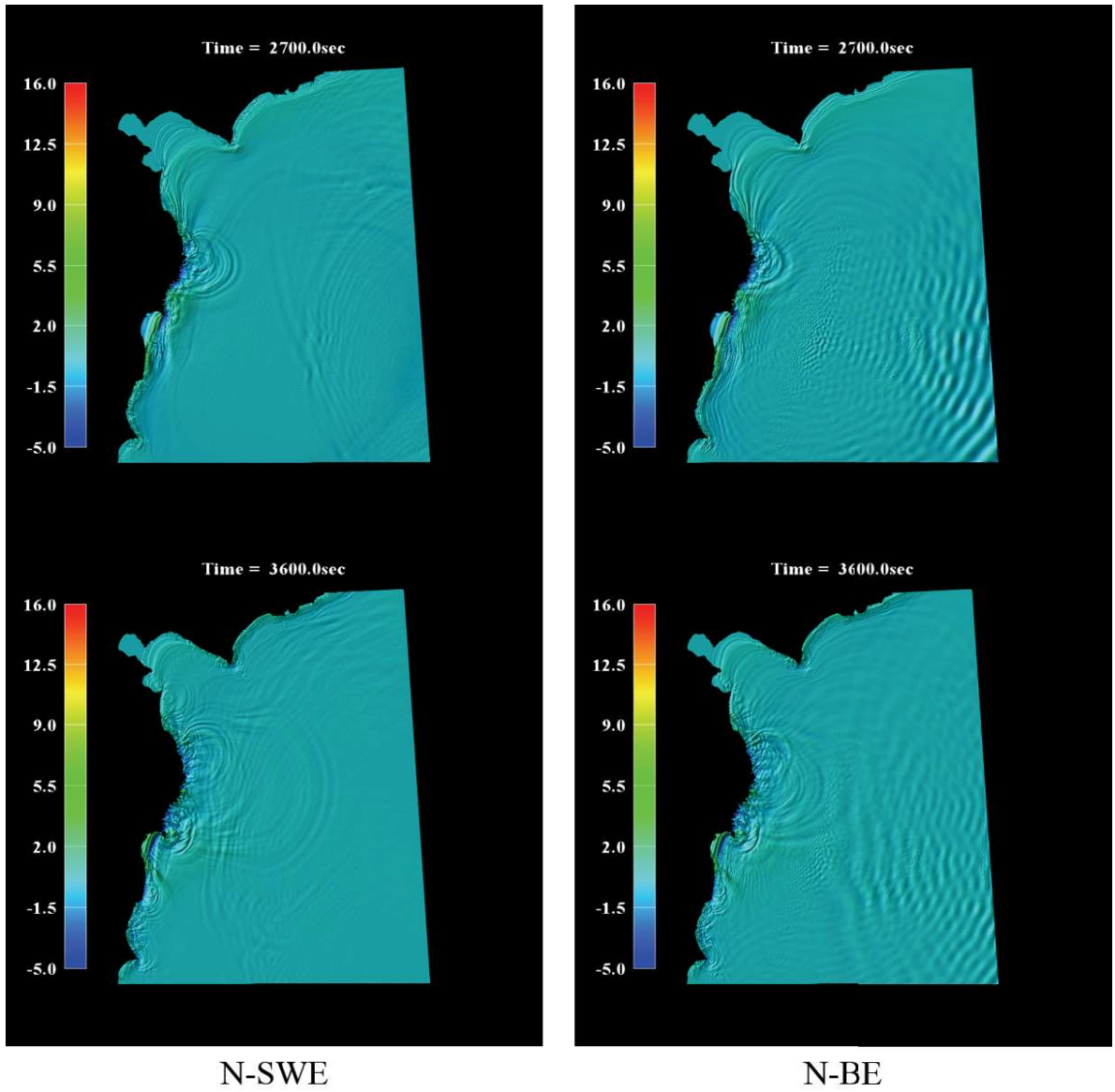
Figure 2.18   The wave propagation at t=700s, 1400s

N-SWE N-BE

Figure **2.19** **The wave propagation at t=2700s, 3600s**

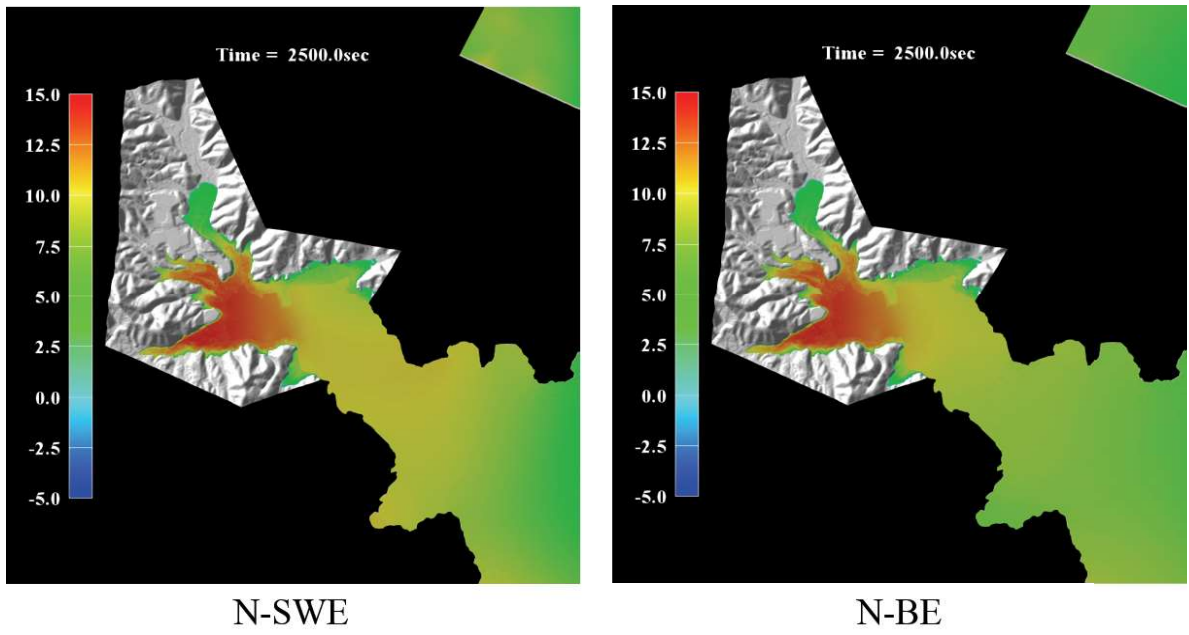N-SWE   N-BE

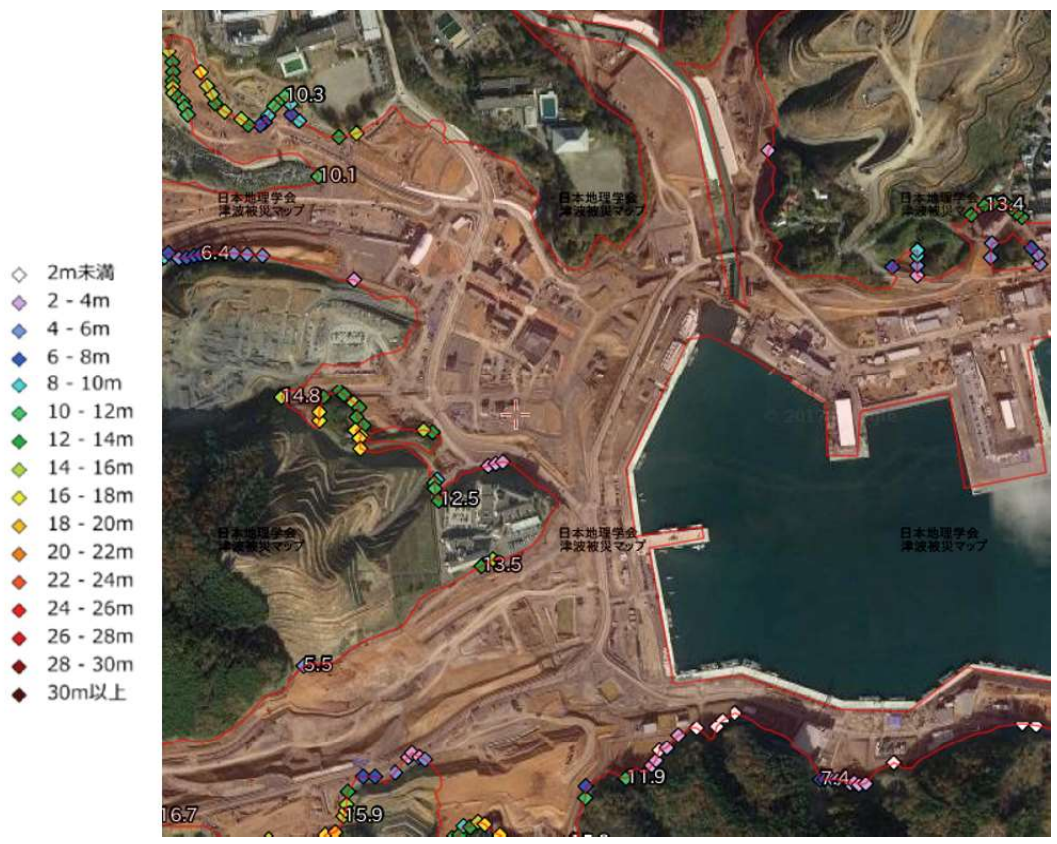Figure 2.20   Inundation areas of computational results at t=2500s

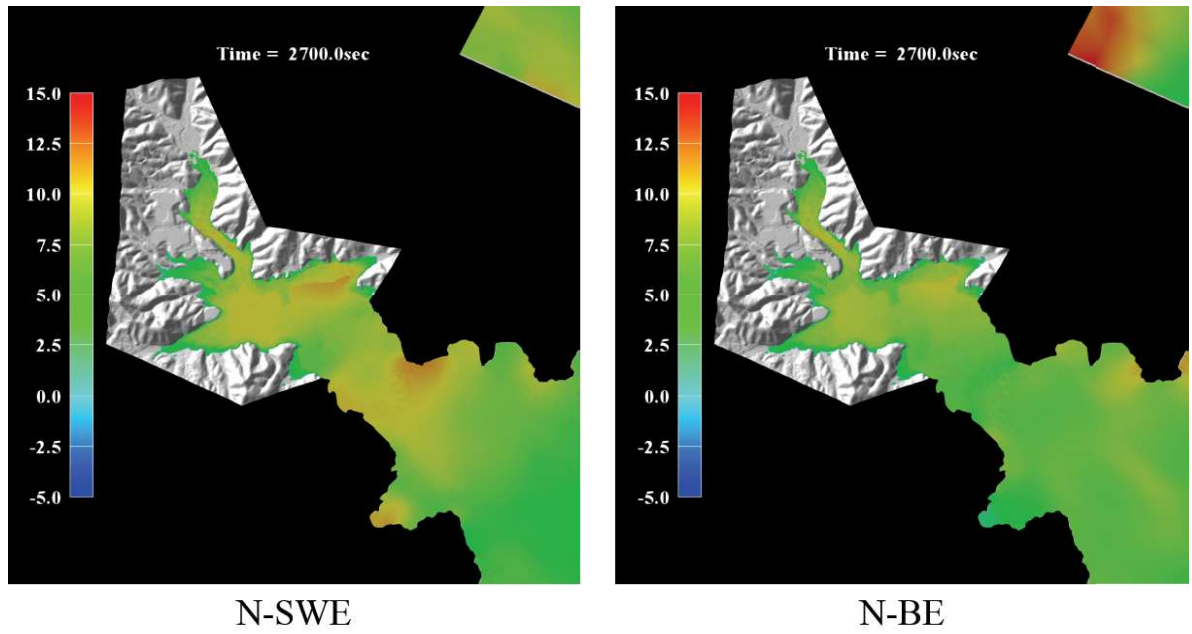Figure 2.21   Runup height by observation[*1]

N-SWE N-BE

Figure 2.22 Inundation area by analysis at t=2700s

Figure 2.23 Inundation area by observation[*1]

2.7.4   Visualization Using AR Technology

Figure **2.24**   **Visualization area by AR technology**

In this study, in order to help improving the quality of education for tsunami disaster prevention, we have used the AR (Augmented Reality) technology to visualize the numerical results of the tsunami caused by the 2011 Great East Japan Earthquake in a local area of Onagawa Town (see **Figure 2.24**). The numerical results are got from the former section.

The visualized results are shown in the **Figure 2.25**. We can see the numerical results are combined with the real space, which can help us to feel the tsunami coming.
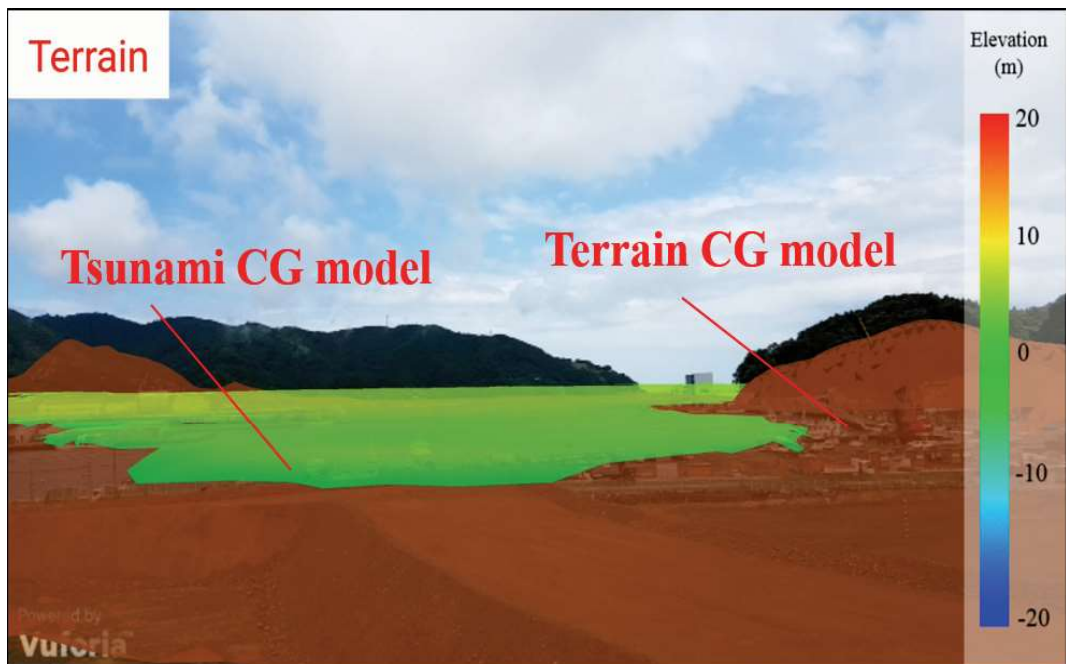
(a) Showing terrain CG model

(b) Hiding terrain CG model

Figure 2.25   **Visualized result by AR technology**

2.8   Chapter Summary

In this chapter, the 2D numerical models using the shallow water equations and the Boussinesq equations were developed. The stabilized finite element method was applied to the spatial discretization. Comparisons were made between the linear/nonlinear shallow water equations and Boussinesq equations. By simulating the numerical examples, the following remarks can be concluded:

- In the investigation of the characteristic for the wave propagating in a slope, the wave attenuation of N-BE is the least, and the result is the best in agreement with the experimental results in the numerical results.
- For the application to the wave-making problem, both of the N-SWE and the N-BE can get good results on the water elevations in comparison to the experiment, even though the N-BE shows less wave attenuation.
- From the large-scale tsunami runup simulation caused by the 2011 Great East Japan Earthquake, the inundation depth and the inundation area of the simulations by using N-SWE and N-BE have shown quite similar to the survey maps.

From the above remarks, the effectiveness and the applicability of the present 2D numerical models using the N-SWE or the N-BE have been confirmed.