Optimal placement of tsunami sensors with depth constraint

Ikha Magdalena[1], Raynaldi La'lang[1], Renier Mendoza[2] and
Jose Ernie Lope[2]

[1] Faculty of Mathematics and Natural Sciences, Institut Teknologi Bandung, Bandung, West Java, Indonesia
[2] Institute of Mathematics, University of the Philippines Diliman, Quezon City, Philippines

ABSTRACT

Tsunamis are destructive natural disasters that can cause severe damage to property and the loss of many lives. To mitigate the damage and casualties, tsunami warning systems are implemented in coastal areas, especially in locations with high seismic activity. This study presents a method to identify the placement of near-shore detection sensors by minimizing the tsunami detection time, obtained by solving the two-dimensional shallow water equations (SWE). Several benchmark tests were done to establish the robustness of the SWE model, which is solved using a staggered finite volume method. The optimization problem is solved using particle swarm optimization (PSO). The proposed method is applied to different test problems. As an application, the method is used to find the optimal location of a detection sensor using data from the 2018 Palu tsunami. Our findings show that detection time can be significantly reduced through the strategic placement of tsunami sensors.

Subjects Optimization Theory and Computation, Scientific Computing and Simulation
Keywords Shallow water equations, Near-shore detection sensors, Particle swarm optimization

INTRODUCTION

Tsunami waves are, in most cases, generated by a seabed deformation caused by some tectonic plate movement, which leads to a sudden displacement of water that travels from the ocean to the shore. These waves may also be generated by landslides, submarine volcanic eruptions, and meteorological disturbances (*Joseph, 2011*; *Okal & Synolakis, 2015*). Unlike tidal waves, they characteristically have longer wavelength and amplitudes that start relatively small but builds up and undergoes shoaling as they reach shallower areas.

In the past 5 years alone, there have been five major tsunami occurrences: in 2016 in Kaikoura, New Zealand (*Heidarzadeh & Satake, 2017*), in 2017 in Greenland (*Chao et al., 2018*), in 2018 in Sulawesi, Indonesia (*Heidarzadeh, Muhari & Wijanarto, 2018*), in 2019 in Sunda Strait, Indonesia (*Grilli et al., 2019*), and in 2020 in the Aegean Sea (*Triantafyllou et al., 2021*). The most devastating was the earthquake-triggered-with-underwater-landslide tsunami in Sulawesi, which killed more than 1,000 and injured over 600. These regions are among those with the highest tsunami risk due to their proximity to tectonic plate boundaries. To minimize the damage and casualties of tsunamis in these high-risk areas, it is necessary to have carefully designed countermeasures, such as tsunami warning systems.

Submitted 12 May 2021
Accepted 1 August 2021
Published 29 September 2021

Corresponding author
Ikha Magdalena,
ikha.magdalena@math.itb.ac.id

Academic editor
Mehmet Cunkas

Additional Information and Declarations can be found on page 25

DOI **10.7717/peerj-cs.685**

Seismic-centered tsunami warning systems typically work in three stages (UN-ESCAP, 2009). As the seismic network detects seismic waves, it sends a signal to seismologists in the warning center. This data is then analyzed to assess whether or not the seismic waves have the potential to generate a tsunami, including the threat level. The responsible party then issues the warning to the public and decides on which action to take. While the exact chain of actions may differ from one country to another, it still follows these general steps (Murjaya, 2012; Doi, 2003).

An example of a tsunami warning system is the Deep-ocean Assessment and Reporting of Tsunamis (DART) (Mungov, Eble & Bouchard, 2012; Bernard & Meinig, 2011; Paros et al., 2011). It comprises three parts: bottom pressure recorders (BPRs), surface buoys, and satellites. Each BPR is anchored to the seafloor to record barometric pressure and temperature. The data are then logged to the buoy as the BPRs read the average water level. Finally, the data are transmitted to the warning center by satellite. Another type of tsunami warning system are near-shore gauges that are used in alerting high-risk coastal communities in case of local tsunamis (Satake, 2014; E2S Warning Signals, 2021). These systems include wave gauges that use ultrasonic waves and buoy and measure offshore sea levels at water depths of 50 to 200 m (Satake, 2014).

Numerous studies on tsunami warning systems have been undertaken by researchers from countries located in the Pacific ring of fire. Mulia & Satake (2020) present the evolution of past to present tsunami observing systems available in Japan. The optimal placement of sensors in Korea was studied by Lee, Jung & Shin, 2020a, Lee, Jung & Shin 2020b. Meanwhile, the use of the rainfall optimization algorithm in the placement of sensors in the Cotabato Trench, Philippines was investigated by Ferrolino, Lope & Mendoza (2020), and an integrated tsunami forecast and warning system, called SIPAT, has been developed and proven successful in Chile (Catalan et al., 2020).

A warning system's performance depends on the ability to optimize the function of each subsystem, and there are many aspects to look at. There have also been numerical methods developed to reproduce the tsunami generation and propagation (Behrens, 2010; Cecioni et al., 2014). Wang & Li (2008) introduced a tsunami warning system that utilizes remote sensing and geographical information systems in monitoring, forecasting, detection, loss evaluation, and relief management. Mulia et al. (2020) proposed an enhanced detection system using airborne platforms, and a 3D topology design for underwater sensor networks was proposed by Lohan, Dube & Agrawal (2020).

Several researchers have proposed methods of optimizing the location of buoys and gauges. Lee, Jung & Shin (2020a) used the Cornell Multi-grid Coupled Tsunami (COMCOT) numerical model and a probabilistic approach to gauge the optimal region for tsunami detection instruments in the eastern sea of Korea. Meza, Catalán and Tsushima (2019) implemented an inversion algorithm to determine the optimal array configuration of offshore tsunami sensors for near-field tsunami forecasting based on three tsunami parameters: arrival time, maximum tsunami amplitude, and forecast skill. Navarrete et al. (2020) used empirical orthogonal function analysis together with a heuristic optimization technique to find the optimal locations of a network of tsunameters. Recently, Wu, Chen & Ghattas (2021) presented a fast and scalable computational framework for finding sensor locations to maximize the expected information gain for a predicted quantity of interest. These results, however, are not quite general as they are calibrated to fit specific bathymetric profiles, tsunami characteristics, or detection sensors. *Ferrolino et al. (2020)* used two-dimensional shallow water equations (SWE) to compute the travel time of tsunami waves and particle swarm optimization (PSO) to find the optimal placement of tsunami sensors. The waves were made to propagate over a wet bed in their simulations, and no particular constraints were specified. However, in real-life scenarios, several restrictions must be satisfied in configuring the detection sensors. For example, near-shore warning systems must be installed in the shallower portion of the sea. To make the system cost-effective, the gauge must be placed within a prescribed water depth and on locations where the tsunami can be detected at the earliest possible time.

The objective of this study is to address the limitations in the simulations done by *Ferrolino et al. (2020)*. First, we use nonlinear SWE with a wet-dry procedure in modeling the tsunami wave propagation so that the model also applies to a water domain with islands. Second, we modify the optimization problem that calculates the optimal placement of sensors by adding a constraint on water depth. This can be used in finding the strategic locations of near-shore tsunami sensors. For example, in the Philippines, tsunami early warning systems are installed near the shore (*DOST, DOST-PHILVOLCS, DOST-ASTI, 2021*).

The numerical solution to the two-dimensional SWE was obtained using the finite volume method (FVM) on a staggered grid, which has been shown to be robust, accurate, and inexpensive (*Pudjaprasetya & Magdalena, 2014*; *Magdalena, Rif'atin & Reeve, 2020*; *Magdalena, Erwina & Pudjaprasetya, 2015*). The arising sensor location problem is a nonlinear programming because of the constraints. We use a penalty method to transform the nonlinear programming problem into an unconstrained optimization problem for this work. Since the resulting cost function is not smooth, we employ PSO, an evolutionary optimization algorithm that does not require the computation of gradients. It was initially formulated to simulate social behavior in colonies (*Kennedy & Eberhart, 1995*), but has since then been explored to solve problems in broader fields, including product design and manufacturing (*Zhou et al., 2006*), smart antenna (*Wagih, Elkamchouchi & Lazinica, 2009*), and of course tsunami detection (*Ferrolino et al., 2020*).

Following this introductory section are three parts. The following section discusses the numerical solution of the SWE using FVM on a staggered grid and the constrained optimization problem. The third section showcases the robustness and accuracy of the proposed algorithm to solve the fluid dynamics model numerically. The section also features our implementation of the PSO algorithm to solve the problem of tsunami detection, first applied to several test problems and finally to actual data obtained from the 2018 Palu tsunami incident. The final section contains our concluding remarks.

METHODS

Fluid dynamics model

This section discusses the SWE, one of many models in fluid dynamics that can be used to model fluid motions. Due to their required assumptions, they are especially accurate in

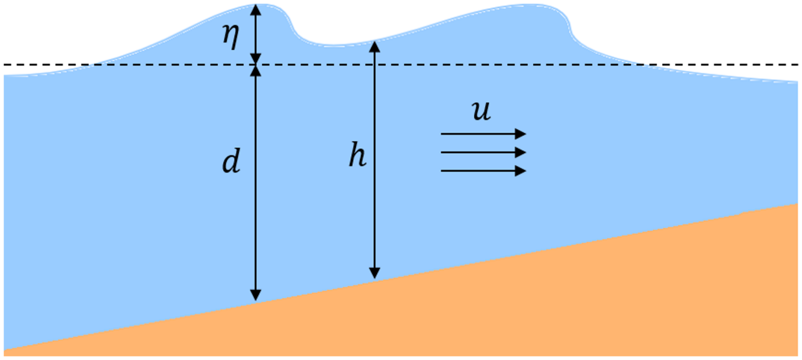

Figure 1 **Illustration of variables used in the 1D SWE.** Full-size DOI: 10.7717/peerj-cs.685/fig-1

solving problems involving the ocean (*Briganti & Dodd, 2009*; *Brocchini & Dodd, 2008*; *Le Roux, Staniforth & Lin 1998*). They can also be used to simulate different tsunami waves, as done by *Geyer & Quirchmayr (2017)* and *Zergani, Aziz & Viswanathan (2015)*.

The SWE arise from a special case of the Navier–Stokes equations, which describe the conservation of mass and momentum. In the case of the SWE, the wavelength of the disturbance is assumed to be long, relative to the depth of the flow. Due to this assumption, the horizontal velocity along a vertical column can be assumed to be uniform, hence it is only a function of the horizontal spatial parameters, namely x and y.

Shallow water equations

The equations in one dimension are obtained by taking a vertical cross-section (in this case, a y-plane cross-section) of some two-dimensional domain and neglecting flows not parallel to the cross-section. Assuming uniform fluid density, incompressible-irrotational flow, and frictionless bed, the set of equations in one dimension is

$$h_t + (hu)_x = 0, \tag{1}$$
$$u_t + g\eta_x + uu_x = 0, \tag{2}$$

where h is the total water depth, u is the horizontal velocity, and g is the gravitational acceleration (throughout this study, the value of g is set to 9.81 m.s^{-2}). We also use the variables d for water depth and η for the free surface elevation, giving the relation $h = d + \eta$. Here, Eq. (1) represents the conservation of mass, while Eq. (2) represents the momentum balance. Figure 1 illustrates the variables just described.

To extend the model to two dimensions, we will have two equations for momentum balance in x- and y-direction. We would then need one more variable: v to denote the horizontal velocity in the y-direction. We also introduce the advection terms vu_y and uv_x in the momentum-conservation-equations. Thus the 2D governing equations read as:

$$h_t + (hu)_x + (hv)_y = 0, \tag{3}$$
$$u_t + g\eta_x + uu_x + vu_y = 0, \tag{4}$$
$$v_t + g\eta_y + vv_y + uv_x = 0. \tag{5}$$

All the variables used in the two-dimensional SWE are illustrated in Fig. 2.

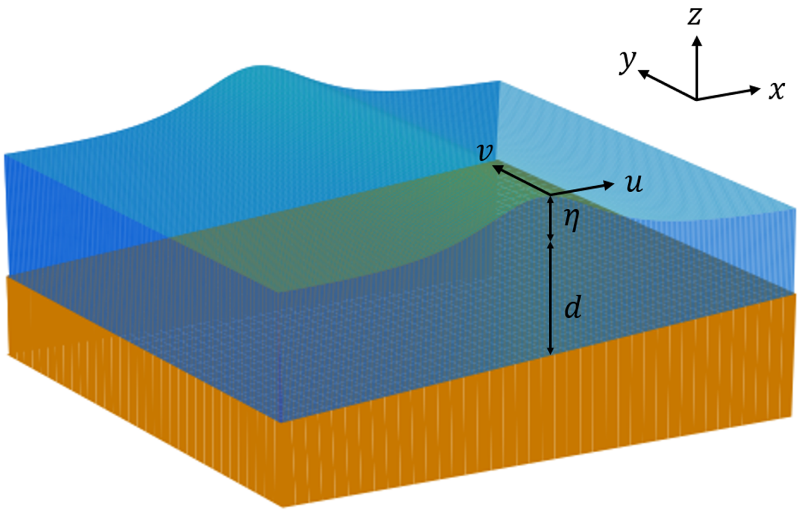

Figure 2 **Illustration of variables used in the 2D SWE.** Full-size DOI: 10.7717/peerj-cs.685/fig-2

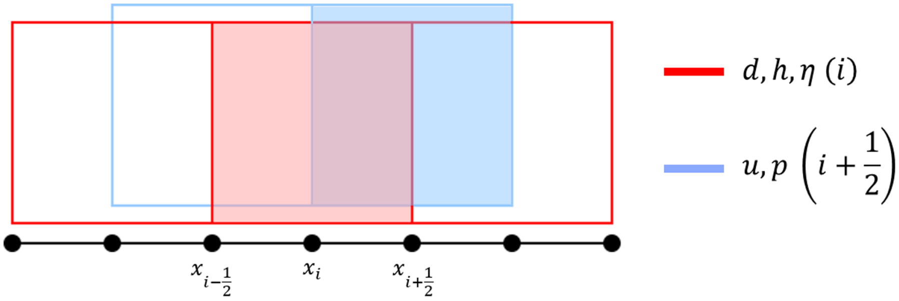

Figure 3 **Staggered grid for the FVM scheme on the 1D SWE.** Full-size DOI: 10.7717/peerj-cs.685/fig-3

Finite volume method on a staggered grid

We apply the finite volume method on a staggered grid to solve the SWE numerically. In a staggered grid arrangement, η is stored in the cell centre, while vectors u and v are stored in the cell faces. We use a structured rectangular grid to partition the spatial domain into intervals of equal size Δx in one dimension, or rectangles of equal size $\Delta x \times \Delta y$ in two dimensions; integer indices (x_i, y_j) represent the cell centers while half-integer indices $(x_{i+\frac{1}{2}}, y_{j+\frac{1}{2}})$ represent the cell faces. These are visualized in Figs. 3 and 4. We denote by t_n the discretization of the time dimension.

Note that computing the spatial derivatives of $p = hu$ and $q = hv$ is not as straightforward, as they are products of two variables stored using two different indices. We approximate the values using the following upwind scheme to compute h at the cell faces, based on the direction of the flow:

$$*h_{i+\frac{1}{2}} = \begin{cases} h_i, & u_{i+\frac{1}{2}} > 0, \\ h_{i+1}, & u_{i+\frac{1}{2}} \leq 0. \end{cases} \tag{6}$$

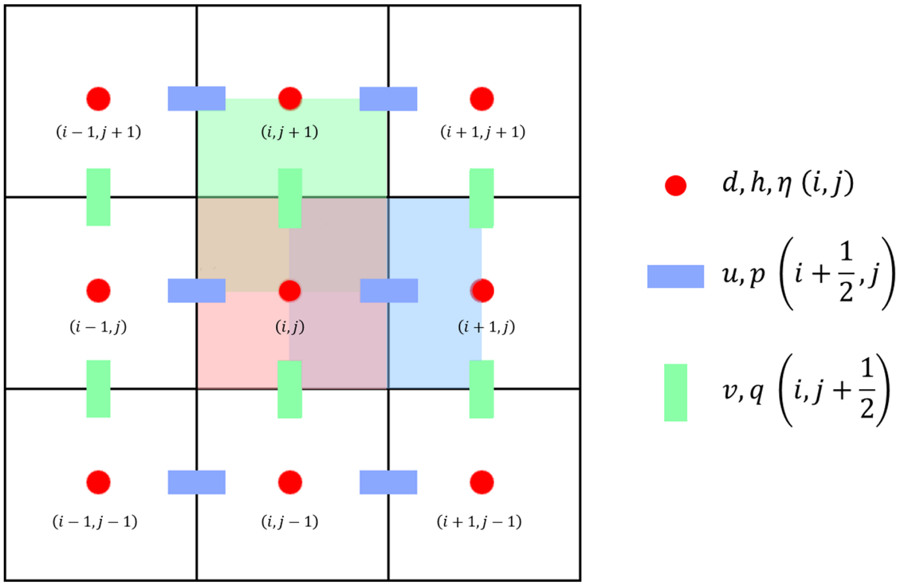

Figure 4 **Staggered grid for the FVM scheme on the 2D SWE.**
Full-size DOI: 10.7717/peerj-cs.685/fig-4

We then define $*p_{i+\frac{1}{2}} = *h_{i+\frac{1}{2}} u_{i+\frac{1}{2}}$. Now, to compute $(uu_x)^n_{i+\frac{1}{2}}$, we take note of the following equality:

$$uu_x = \frac{p}{h} u_x. \tag{7}$$

This is a modification of the method earlier used by *Magdalena, Erwina & Pudjaprasetya (2015)* and is preferred here because the calculation of the advection term is simpler and less costly. Next, the value of h at half-integer indices and the value of p and u at integer indices are computed as

$$\overline{h}_{i+\frac{1}{2}} = \tfrac{1}{2}(h_i + h_{i+1}), \tag{8}$$

$$\overline{p}_i = \tfrac{1}{2}\left(p_{i+\frac{1}{2}} + p_{i-\frac{1}{2}}\right), \tag{9}$$

$$*u_i = \begin{cases} u_{i-\frac{1}{2}}, & \overline{p}_i > 0 \\ u_{i+\frac{1}{2}}, & \overline{p}_i \le 0. \end{cases} \tag{10}$$

These quantities yield the following approximation of $(uu_x)^n_{i+\frac{1}{2}}$:

$$(uu_x)_{i+\frac{1}{2}} = \frac{\overline{p}_{i+\frac{1}{2}}}{\overline{h}_{i+\frac{1}{2}}} \left(\frac{*u_{i+1} - *u_i}{\Delta x}\right), \tag{11}$$

Thus, we obtain this discretization of the one-dimensional SWE:

$$\frac{h_i^{n+1}-h_i^n}{\Delta t}+\frac{{*p}^n_{i+\frac{1}{2}}-{*p}^n_{i-\frac{1}{2}}}{\Delta x}=0, \tag{12}$$

$$\frac{u^{n+1}_{i+\frac{1}{2}}-u^n_{i+\frac{1}{2}}}{\Delta t}+g\frac{\eta^{n+1}_{i+1}-\eta^{n+1}_i}{\Delta x}+(uu_x)^n_{i+\frac{1}{2}}=0. \tag{13}$$

Extending the finite volume method on SWE to two dimensions require us to compute $(vu_y)^n_{i+\frac{1}{2},j}$ $(vv_y)^n_{i,j+\frac{1}{2}}$ and $(uv_x)^n_{i,j+\frac{1}{2}}$. We only give the details in approximating the first expression, as the others can be done similarly. This time, we use the equalities

$$vu_y=\frac{q}{h}u_y,\qquad vv_y=\frac{q}{h}v_y,\qquad \mathit{and}\qquad uv_x=\frac{p}{h}v_x. \tag{14}$$

Analogous to the first advection term, we do an approximation for the following variables:

$$\bar{h}_{i+\frac{1}{2},j}=\tfrac{1}{2}\left(h_{i,j}+h_{i+1,j}\right), \tag{15}$$

$$\bar{q}_{i+\frac{1}{2},j+\frac{1}{2}}=\tfrac{1}{2}\left(q_{i+1,j+\frac{1}{2}}+q_{i,j+\frac{1}{2}}\right), \tag{16}$$

$$*u_{i+\frac{1}{2},j+\frac{1}{2}}=\begin{cases} u_{i+\frac{1}{2},j}, & \bar{q}_{i+\frac{1}{2},j+\frac{1}{2}}>0\\ u_{i+\frac{1}{2},j+1}, & \bar{q}_{i+\frac{1}{2},j+\frac{1}{2}}<0.\end{cases} \tag{17}$$

Thus, we have these approximations:

$$(vu_y)_{i+\frac{1}{2},j}=\frac{\bar{q}_{i+\frac{1}{2},j}}{\bar{h}_{i+\frac{1}{2},j}}\left(\frac{*u_{i+\frac{1}{2},j+\frac{1}{2}}-*u_{i+\frac{1}{2},j-\frac{1}{2}}}{\Delta y}\right), \tag{18}$$

$$(vv_y)_{i,j+\frac{1}{2}}=\frac{\bar{q}_{i,j+\frac{1}{2}}}{\bar{h}_{i,j+\frac{1}{2}}}\left(\frac{v_{i,j+\frac{1}{2}}-v_{i,j-\frac{1}{2}}}{\Delta y}\right), \tag{19}$$

$$(uv_x)_{i,j+\frac{1}{2}}=\frac{\bar{p}_{i,j+\frac{1}{2}}}{\bar{h}_{i,j+\frac{1}{2}}}\left(\frac{*v_{i+\frac{1}{2},j+\frac{1}{2}}-*v_{i+\frac{1}{2},j-\frac{1}{2}}}{\Delta x}\right), \tag{20}$$

and, finally, we obtain this discretization of the two-dimensional SWE:

$$\frac{h^{n+1}_{i,j}-h^n_{i,j}}{\Delta t}+\frac{{}^*p^n_{i+\frac{1}{2},j}-{}^*p^n_{i-\frac{1}{2},j}}{\Delta x}+\frac{{}^*q^n_{i,j+\frac{1}{2}}-{}^*q^n_{i,j-\frac{1}{2}}}{\Delta y}=0, \tag{21}$$

$$\frac{u^{n+1}_{i+\frac{1}{2},j}-u^n_{i+\frac{1}{2},j}}{\Delta t}+g\frac{\eta^{n+1}_{i+1,j}-\eta^{n+1}_{i,j}}{\Delta x}+(uu_x)^n_{i+\frac{1}{2},j}+(vu_y)^n_{i+\frac{1}{2},j}=0, \tag{22}$$

$$\frac{v^{n+1}_{i,j+\frac{1}{2}} - v^{n}_{i,j+\frac{1}{2}}}{\Delta t} + g\frac{\eta^{n+1}_{i,j+1} - \eta^{n+1}_{i,j}}{\Delta y} + \left(vv_y\right)^{n}_{i,j+\frac{1}{2}} + \left(uv_x\right)^{n}_{i,j+\frac{1}{2}} = 0. \tag{23}$$

Proposed approach in Tsunami sensor placement

The sensors should be placed in locations where detection time is minimized. We now discuss the optimization formulation arising from this real-world application. We modify the minimization problem proposed by *Ferrolino et al. (2020)*, *Ferrolino, Lope & Mendoza (2020)* by adding a constraint on the water depth.

Optimization problem

Suppose we have source points $R = \{r_1, \ldots, r_k\}$, where tsunami waves are generated. Let s be an arbitrary point in the spatial domain. We define

$$\begin{aligned}\tau(r_i; s) :&= \text{the time it takes for a wave generated from a source point} \\ &\quad r_i \text{ to reach the point } s.\end{aligned} \tag{24}$$

To compute $\tau(r_i; s)$, we use the two-dimensional SWE presented previously. Suppose we wish to place l sensors, $\sigma = \{s_1, \ldots, s_l\}$. Since only one sensor is needed to be triggered for the tsunami to detected, the time it takes for the tsunami to be detected from a source r_i is given by

$$t(r_i; \sigma) := \min_{1 \le j \le l} \tau\left(r_i; s_j\right).$$

Thus, the guaranteed tsunami detection time to reach at least one of the sensors $\sigma = \{s_1, \ldots, s_l\}$ from any of the possible sources is given by

$$T(\sigma) = \max_{1 \le i \le k} t(r_i; \sigma). \tag{25}$$

Suppose the sensors can be placed anywhere in the water domain D, then the goal is to minimize (25). This minimization problem was also used in previous studies (*Ferrolino, Lope & Mendoza, 2020*, *Ferrolino et al., 2020*; *Astrakova et al., 2009*).

In this work, we introduce a constraint by considering the scenario where the sensors can only be placed on locations with depth constraint, as in the case of near-shore tsunami sensors. If we denote by D^* the points in D that are within the depth constraint (see Fig. 5), then the minimization problem we consider is given by

$$\begin{cases} \quad \min\limits_{\sigma} \; T(\sigma), \\ \text{subject to } \sigma \in D^*. \end{cases} \tag{26}$$

The constrained domain D^* can be defined as $D^* = \{(x, y) \in D \colon d(x, y) \le d_{max}\}$, where d_{max} is the maximum depth in which a sensor can be placed. The constrained optimization problem (26) can be reformulated as an unconstrained problem using the penalty method, that is, by considering the problem

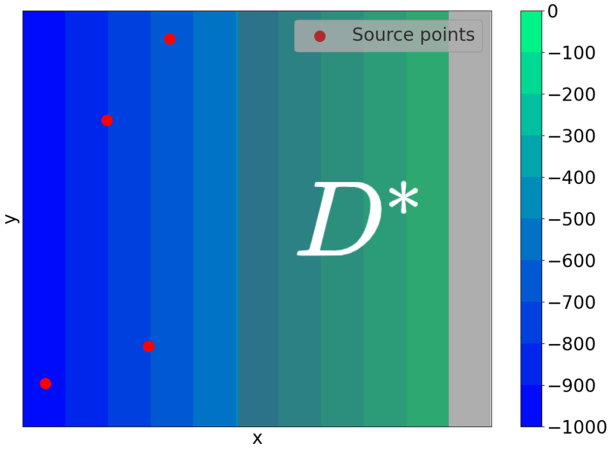

Figure 5 This is an example of a domain D. The tsunami sources are shown in red and the set of points that are within the depth constraint is labeled D^*.

Full-size DOI: 10.7717/peerj-cs.685/fig-5

$$\min_{\sigma} T(\sigma) + \mu\chi_{D\backslash D^*}(\sigma), \tag{27}$$

where μ is the penalty parameter and $\chi_{D\backslash D^*}(\sigma)$ is the characteristic function (or indicator function) of $D\backslash D^*$, whose value is 0 if $\sigma \in D^*$ and 1 otherwise. We set μ equal to a large number to penalize those sensor locations that do not satisfy the depth constraint. In our simulations, we set $\mu = 10^6$.

It should be noted that this formulation can be easily modified to consider cases when the tsunami does not originate from a point source (*e.g.*, when the tsunami is due to a landslide). The minimization problem remains the same but the initial conditions of the SWE will have to be modified. Likewise, this approach can also be used to determine the location of deep-ocean tsunami sensors by either setting the depth constraint to a high value or the value of μ to 0.

Particle swarm optimization

As discussed previously, the function τ in (24) is computed by solving the two-dimensional SWE numerically. However, this only gives us the values of τ at points located at the cell centers of the rectangular grid. To calculate the travel time at other locations, we use bilinear interpolation using the values of τ at the surrounding points.

Bilinear interpolation is done by performing linear interpolation in one direction, and then performing it again in the other direction, on every rectangle $[x_i, x_{i+1}] \times [y_j, y_{j+1}]$. For example, suppose we want to interpolate τ using some function $f(x, y)$ on the unit square $[0, 1] \times [0, 1]$, given its values at the points (0, 0), (1, 0), (0, 1), and (1, 1). Then $f(x, y)$ is given by

$$f(x,y) = f(0,0)(1-x)(1-y) + f(1,0)x(1-y) + f(0,1)(1-x)y + f(1,1)xy, \quad (x,y) \in [0,1]^2. \tag{28}$$

An illustration of this is provided in Fig. 6.

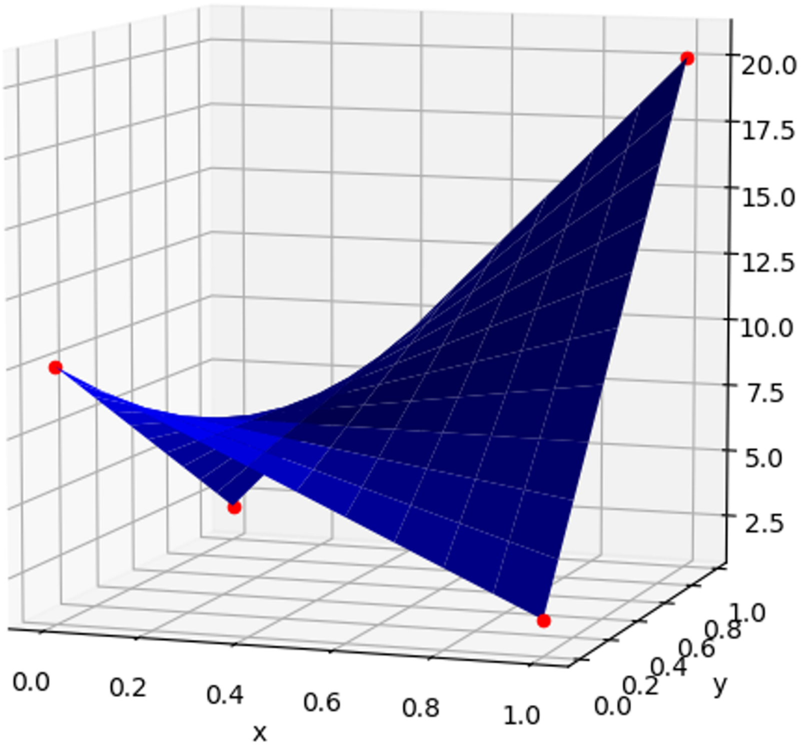

Figure 6 The values of the interpolants (red points) are used to calculate the bilinear interpolation (surface plot) using Equation (28). Full-size DOI: 10.7717/peerj-cs.685/fig-6

Despite its name, as seen above, the resulting interpolant is not linear; rather, it is the sum of products of linear functions in x and y, respectively. Hence, the whole function is piecewise quadratic and may not be differentiable at the cell edges. Furthermore, the constrained optimization problem is transformed into an unconstrained problem using a penalty method. The penalty term in (27) includes a characteristic function, which is binary and not differentiable. In view of this, it is fitting to use optimization methods that do not rely on gradients. In this work, we explore the use of derivative-free metaheuristic algorithms in solving the minimization problem.

Metaheuristic algorithms have gained popularity because they only rely on function evaluations and have the capability of obtaining the global minimizer (*Yang, 2014*). It was shown in *Ferrolino et al. (2020)* how PSO was effective in solving the tsunami location problem without depth constraint. We adopt the same algorithm for our problem formulation.

The PSO algorithm starts by randomly generating a swarm of solution candidates (particles) with their initial velocity vector. Each particle then move around the search-space according to three parameters: its velocity vector, the vector pointing at its own best-known position, and the vector pointing at the swarm's best-known solution. For each iteration, the step that each particle takes is a linear combination of these three vectors (see Fig. 7), expecting that the swarm eventually gathers around one point—the optimal solution.

The pseudocode of the algorithm is presented in Algorithm 1. The algorithm starts by generating $swarm_{size}$ particles randomly in the search space D, and each their own initial

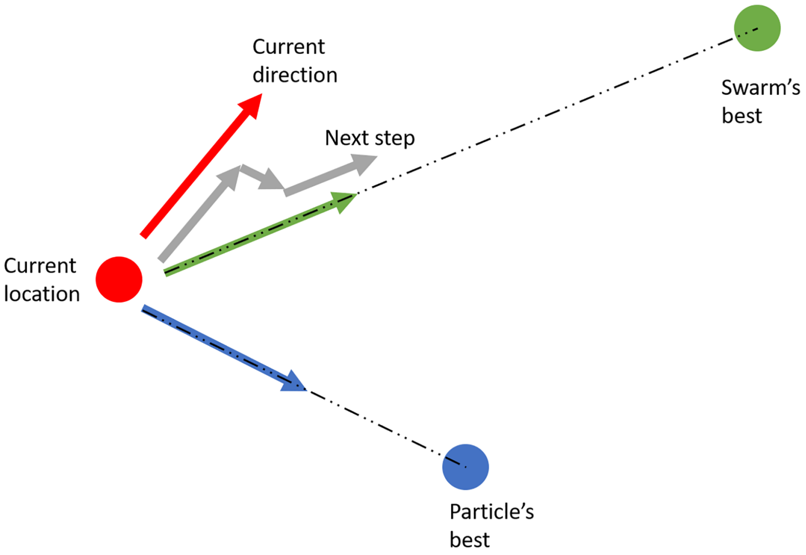

Figure 7 **Illustration of the particle swarm optimization.**
Full-size DOI: 10.7717/peerj-cs.685/fig-7

velocity vector **v**. They all then follow a series of instructions based on computations provided in Algorithm 1, where the vectors **p** (each particle's best known location) and **g** (the entire swarm's best known location) is updated. This process terminates when either the maximum number of iterations (max_{iter}) is reached, or the change in the objective value f is less than min_{func}. In this study, the function f is the one presented in Eq. (26); and the value of each hyperparameters are $swarm_{size} = 100$, $max_{iter} = 100$, $min_{func} = 10^{-8}$, $\omega = \phi_p = \phi_g = 0.5$.

More detailed discussions of this method may be found in *Kennedy & Eberhart (1995)*; *Shi & Eberhart (1998)*, *Mezura-Montes & Coello (2011)*. Our implementation of the PSO algorithm made use of the built-in command *PySwarms*, an extensible research toolkit for particle swarm optimization in *Python*.

RESULTS

Benchmark test and validation of SWE

In order to get an idea of the numerical model's robustness and accuracy, some benchmark tests are performed. The model is considered using different types of initial conditions, boundary conditions, and bottom profiles. These simulations test the performance of the SWE.

1D standing wave on a flat bottom

A standing wave, simply put, is a wave that does not travel (horizontally). Rather, it is an oscillating wave fixed in space. At any given point in space, the peak amplitude of wave oscillations on a standing wave is constant. This phenomenon can be observed on the surface of a liquid in a vibrating container.

To produce a standing wave simulation in one dimension, we set up a container of length L, constant still water depth d; give it initial conditions $\eta(x, 0) = \cos(\frac{\pi x}{L})$, $u(x, 0) = 0$

Algorithm 1 **Particle swarm optimization.**

```
input: $D$ (search space), $f$ (objective function)
output: $g$ (swarm's best known location)
hyperparameters: $swarm_{size}$, $max_{iter}$, $min_{func}$, $\omega$, $\phi_p$, $\phi g$
begin
  let $swarm \leftarrow generateSwarm(swarm_{size})$;
  for each $particle \in swarm$ do set its location as the best known position;
  let $g \leftarrow arg\,min_p f(p)$ $p$: each particle's best known position
  for $i \leftarrow 1$ to $max_{iter}$ do
    for $particle \in swarm$ do
      let $x$, $v$, $p$ as the particle's position, current velocity, best known position;
      let $r_p, r_g \leftarrow random(0, 1)$;
      $v \leftarrow \omega v + r_p\phi_p(p - x) + r_g\phi_g(g - x)$;
      $x \leftarrow x + v$;
      if $f(x) < f(p)$ then $p \leftarrow x$;
    end
    $g \leftarrow arg\,min_p f(p)$;
  end
end
```

and wall boundary conditions $u(0, t) = u(L, t) = 0$. The result of the simulation on a flat bottom with length $L = 25$ m and depth of 4 m, using $\Delta x = 0.2$ m, is shown in Fig. 8. The wave seemingly does not travel horizontally, as a result of interference between two waves traveling in opposite directions. The water level at any point c oscillates in time with amplitude $|\eta(c, 0)|$. The analytical solution to this phenomenon is equivalent to that of a string of infinite length with some wave number k and angular frequency ω. In linear and non-dispersive case here, these numbers are just $k = \frac{\pi}{L}$ and $\omega = k\sqrt{gd}$, which gives

$$\eta(x, t) = \cos\left(\frac{\pi}{L}x\right)\cos\left(\frac{\pi}{L}\sqrt{gd}\,t\right). \tag{29}$$

Simulating this at $\Delta x = 0.2$ m, $\Delta t = 0.032$ s for 20 s gives a mean absolute error of 0.015 m. To simplify things, all simulations below assume hard wall boundary conditions and no current, unless stated otherwise.

1D wave run-up on a sloping beach

The first nonlinear simulation is wave run-up, which refers to either the phenomenon or the measure of when an ocean wave reaches a beach or sea dike structure and rises above still water level. Obviously, as a wave approaches the beach structure, the water depth gets smaller relative to the amplitude, hence the advection term becomes more significant.

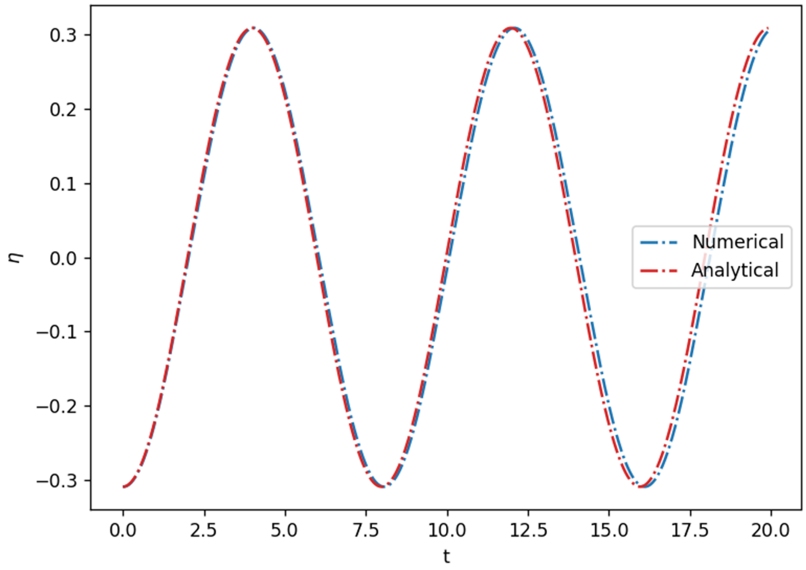

Figure 8 **Comparison between numerical and analytical solutions of the 1D standing wave on a flat bottom, at x = 15 m.** Full-size DOI: 10.7717/peerj-cs.685/fig-8

Furthermore, in this simulation the wave interacts with the dry region, something that was not observed in the previous simulations.

This simulation is based on an experiment by *Synolakis (1987)* on a solitary wave that propagates through a single slope. The solitary wave centered at $x = x_0$ has an initial surface profile $\eta(x,0) = \frac{H}{d} sech^2\left(\gamma\left(x - \frac{L}{4}\right)\right)$ with $\gamma = \sqrt{\frac{3H}{4d}}$. This way, similar to the step bottom simulation, the right traveling soliton eventually changes shape, this time as it climbs the sloping beach. This behavior shown in Fig. 9, where the simulation was done with Δx = 0.1 m and Δt = 0.016 s. Several gauges were placed during the experiment to record the wave height every 10 s. We then compared these records to the simulation. As seen in Fig. 9, the model was able to predict the experiment to a pretty high accuracy.

1D dam break over a flat, dry bed

Dam break, as the name suggests, happens when the dam holding a reservoir breaks, causing a sudden rush of water flowing to the other side. The set up of this simulation is on a flat bed with 1-m-deep water on one side of the domain (reservoir), and completely dry on the other side:

$$h(x,0) = \begin{cases} H, & x < x_0, \\ 0, & x > x_0, \end{cases} \tag{30}$$

where x_0 represents the location of the barrier. Right after the simulation starts, the previously contained water will stream to the dry side and start to fill it, while the water level on the reservoir goes down. The result of the simulation on a 20 m-long, 1 m-deep bed with Δx = 0.1 m is provided in Fig. 10. To test the robustness of the numerical scheme, we compare it with the analytical solution given by *Chanson (2008)*:

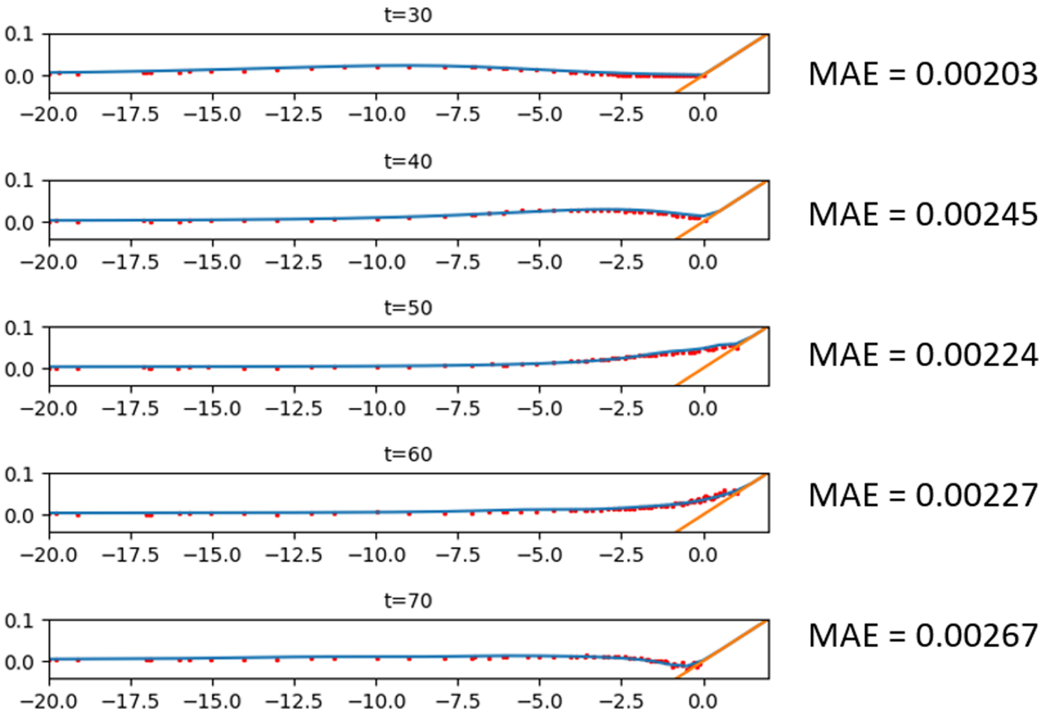

Figure 9 Comparison between Synolakis' experiment and the numerical solution of the 1D wave run-up on a sloping beach. Full-size DOI: 10.7717/peerj-cs.685/fig-9

$$h(x,t) = \begin{cases} H, & x < -ct, \\ \dfrac{H}{9}\left(\dfrac{2-x}{ct}\right)^2, & -ct \le x < (2c-1)t, \end{cases} \tag{31}$$

where $c = \sqrt{gH}$. At Δx = 0.1 m, Δt = 0.008 s, the numerical scheme gives a mean absolute error of 0.0034 m.

2D planar surface wave over a parabolic basin

So far the simulations were set on rectangular or trapezoidal shapes. To avoid bias due to the also rectangular shape of the grids, this simulation will be set on a parabolic basin. This way, not only do we have a curved bottom profile, all of the perimeter will be interacting with dry regions. The results of this simulation further confirms the performance of the wet-dry procedure.

The shape of the basin follows the parabolic function $d(x,y) = d_0(1-(x^2+y^2)/L^2)$ on an 8,000 m × 8,000 m domain, with L^2 = 8,000 m^2. The large parabolic shape of the domain is supposed to mimic that of a lake. We would like to see the motion of the water surface initialized by $\eta(x,y,0) = \frac{2a_0 d_0}{L}\left(\frac{x}{L} - \frac{\xi}{2L}\right)$. The analytical solution of the planar surface wave obtained by *Thacker (1981)* is

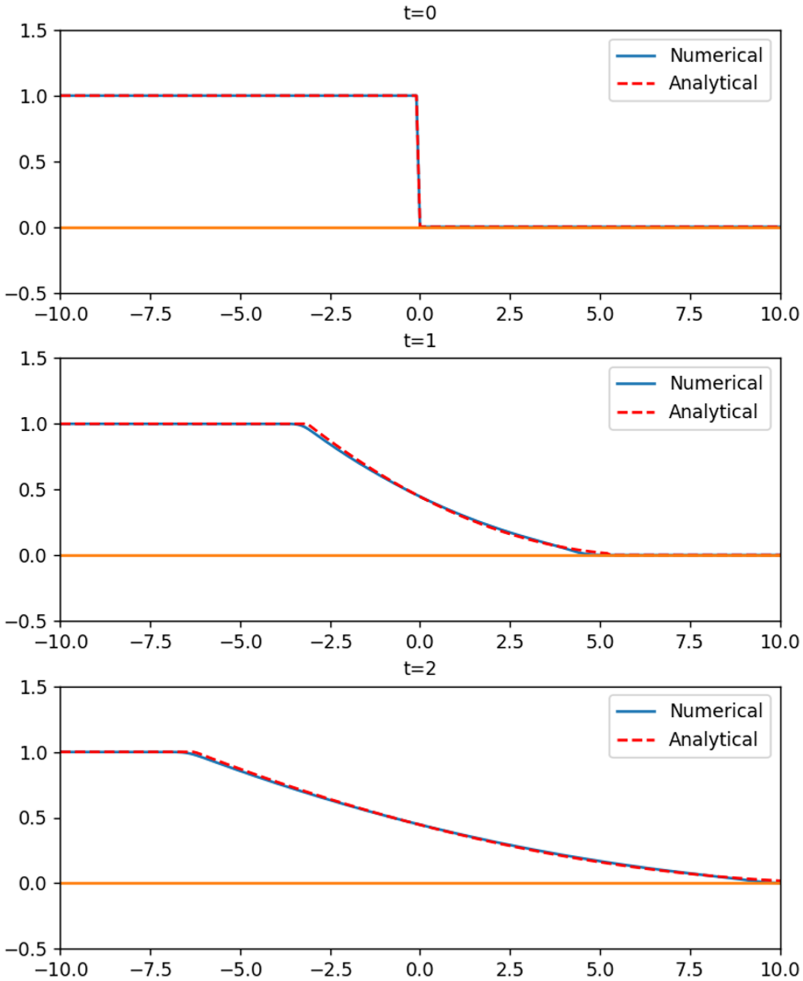

Figure 10 **Comparison between the analytical and numerical solutions of the 1D dambreak over a flat, dry bed.** Full-size DOI: 10.7717/peerj-cs.685/fig-10

$$\eta(x, y, t) = \frac{2\xi d_0}{L}\left(\frac{x}{L}\cos\omega t - \frac{y}{L}\sin\omega t - \frac{\xi}{2L}\right), \quad \omega = \frac{1}{L}\sqrt{2gd_0}. \tag{32}$$

The parameters used in the simulation were $d_0 = 1$ m, $\xi = 400$ m using $\Delta x = \Delta y = 40$ m, and $\Delta t = 4.5$ s. As expected, the planar surface in the parabolic basin oscillates with period $T = 2\pi/\omega = 2\pi L/\sqrt{gd_0}$, which can be seen in Fig. 11. Quantitatively, the numerical solution agrees with the analytical solution with mean absolute error of 0.029 m.

Validation with tsunami-related experiments

Finally, the performance of staggered finite volume scheme for the SWE will be tested on tsunami-related events. To do this, the model is tested on two experiments: “solitary wave on a conical island” and “tsunami run-up onto a complex 3D beach.”

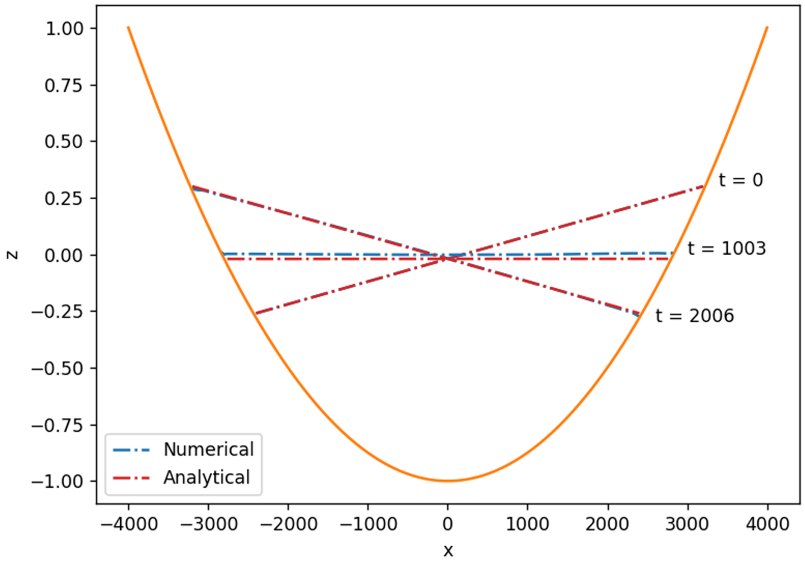

Figure 11 **Cross section along y = 0 of the 2D planar surface wave over a parabolic basin.**
Full-size DOI: 10.7717/peerj-cs.685/fig-11

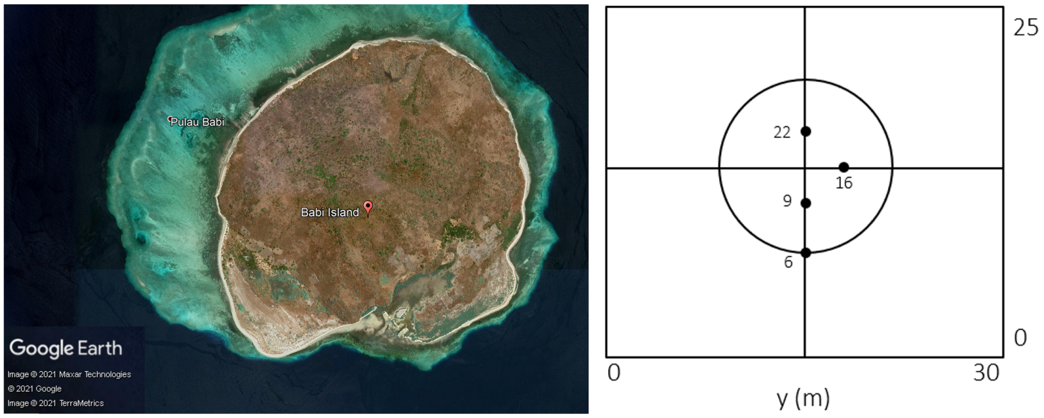

Figure 12 **Satellite view of Babi Island (left); conical island experiment setup (right).**
Full-size DOI: 10.7717/peerj-cs.685/fig-12

Tsunami run-up onto a conical island

This experiment was done at the Coastal Engineering Research Center, Vicksburg, Mississippi, as part of a research (*Briggs et al., 1995*) on a tsunami that struck Babi Island, Indonesia in 1992. The shape of Babi Island is similar to a truncated cone, as seen in Fig. 12, and the experiment may serve as a benchmark test for fluid dynamic models in modeling interaction between waves and such structure. It was conducted on a 25 m × 30 m basin with a conical island at the center, and waves with an initial solitary wave-like profile were generated from one side of the tank. The resulting water surface elevations were measured by 22 gauges. Our numerical simulation was ran with $\Delta x = \Delta y = 0.2$ m. The results of the simulation were compared with data gathered at 4 of the 22 gauges: G6,

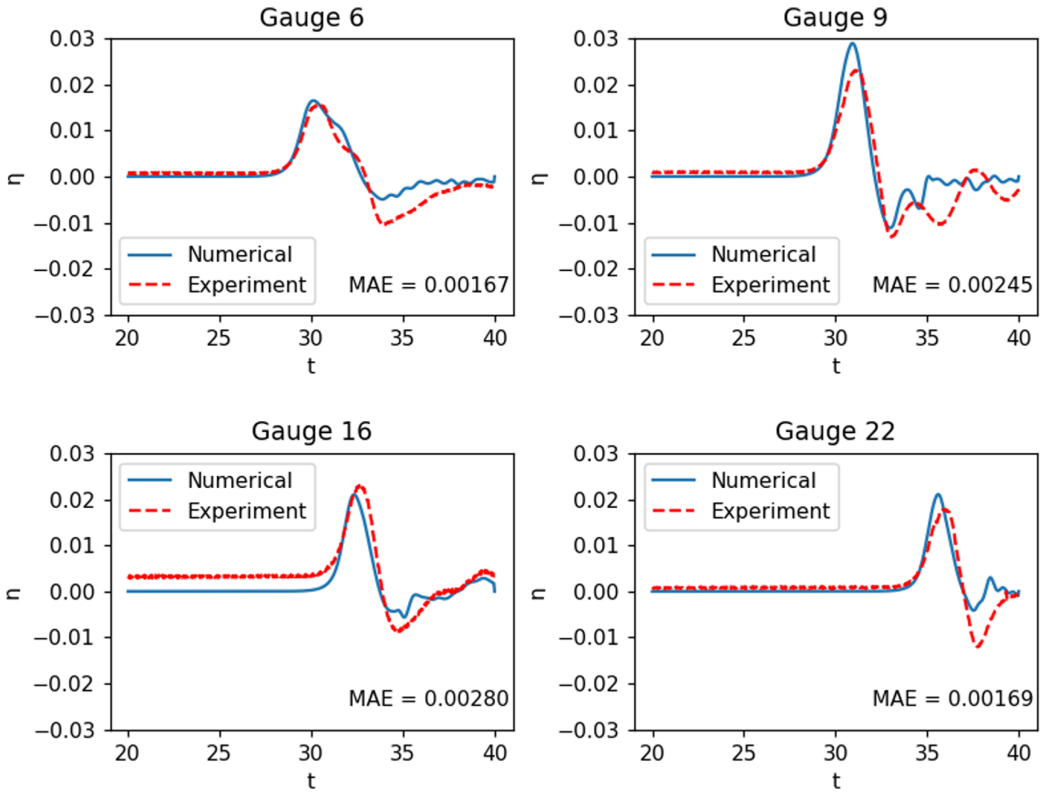

Figure 13 Comparison between the numerical solution of the tsunami run-up onto aconical island and data obtained from the laboratory experiment. Full-size DOI: 10.7717/peerj-cs.685/fig-13

G9, G16, and G22. As seen in Fig. 13, the model was able to replicate the experiment with great accuracy.

Tsunami run-up onto a complex 3D beach

We then move on to an experiment with a much more complex bathymetry, similar to that found in nature. The experiment was based on a tsunami that struck Okushiri Island, Japan, with a really high current, that resulted in an extreme run up at the tip of a very narrow gulley within a small cove at Monai. It was done at the Central Research Institute for Electric Power Industry (CRIEPI) in Abiko, Japan on a 1:400 laboratory model of Monai (*Liu, Yeh & Synolakis, 2008*) and the setup in shown in Fig. 14. Just like the previous experiment, a series of waves were generated from the deeper side of the water. The results of our simulation, with $\Delta x = \Delta y = 0.025$ m, were compared with the recorded height on the three gauges placed during the experiment. As can be observed in Fig. 15, the model was able to replicate the experiment with high accuracy.

OPTIMAL PLACEMENT OF DETECTION SENSORS

Having established the robustness of the shallow-water model, we now solve the optimization problem under different scenarios. This section is divided into three parts.

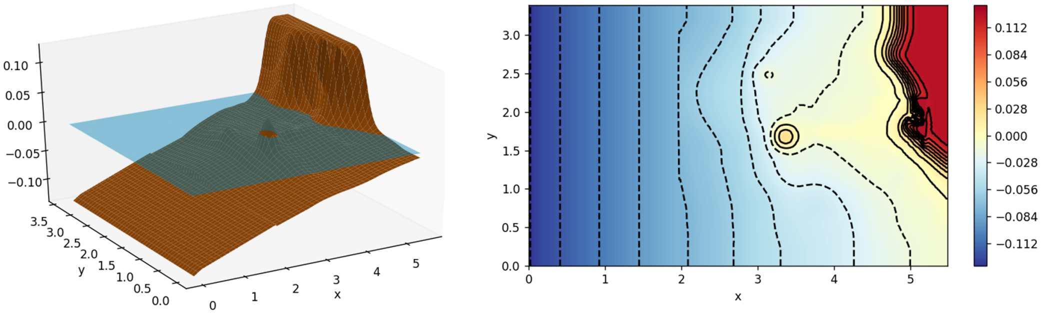

Figure 14 **Monai laboratory model (left); contour of Monai laboratory model (right).**
Full-size DOI: 10.7717/peerj-cs.685/fig-14

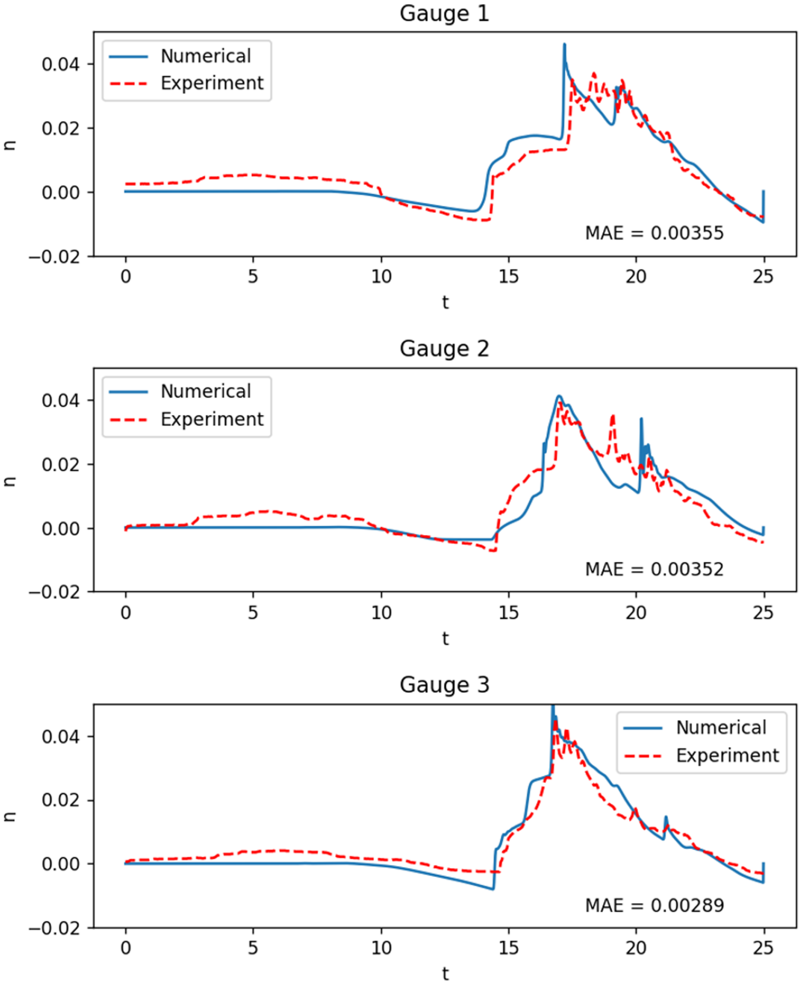

Figure 15 **Comparison between the numerical solution of the tsunami run-up onto a complex 3D beach and data obtained from the laboratory experiment.**
Full-size DOI: 10.7717/peerj-cs.685/fig-15

First, we investigate the case when there is only one sensor available. In the next part, we look at how the first case compares with having multiple detection sensors. Finally, we implement the proposed method using data from the 2018 Palu tsunami incident.

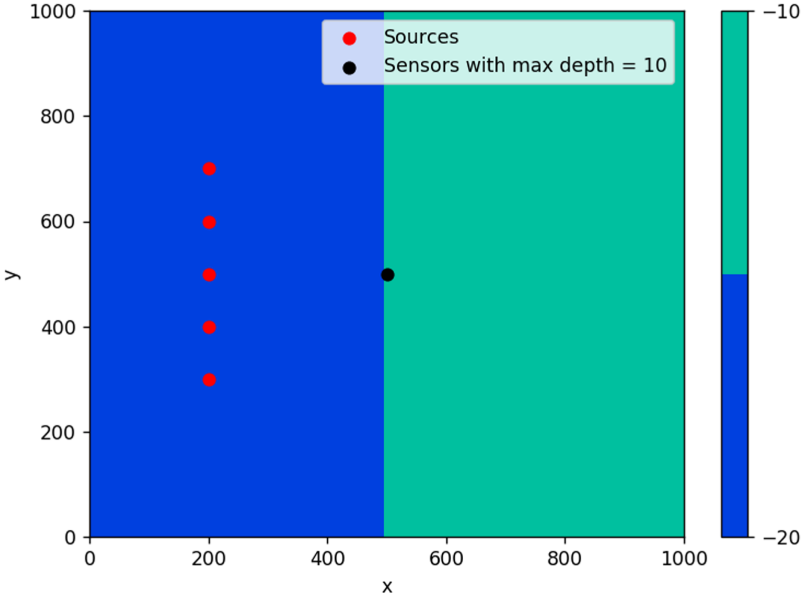

Figure 16 **Optimal location of one detection sensor on a step bottom.**
Full-size DOI: 10.7717/peerj-cs.685/fig-16

One detection sensor

Step bottom

The first simulation was done on a step bathymetry of size 1,000 m × 1,000 m with $d = 20$ m for $x < 500$ m, and $d = 10$ m elsewhere. Waves were generated from five evenly spaced points along the line $x = 200$ m, as seen in Fig. 16. The feasible region for this problem is $\{(x, y): d(x, y) \leq 10 \text{ m}\}$, which is the whole right side of the domain.

The result obtained is pretty intuitive, that is, the optimal location of detection sensor in this case is right in the middle of the line where the domain changes depth. If located far up on the y-axis, it will take long for the sensor to detect waves coming from source points further down (and vice versa). If placed too far to the right, waves from any source point will take a longer time to reach the sensor.

Sloping bottom

The next simulation was done on a sloping bathymetry of size 10,000 m × 10,000 m, with a slope of 0.1 and maximum depth of 1,000 m. This time the waves were generated through a flip-fault motion from six subduction zones in the shape of arcs of the semicircle $x + 5000 = -\sqrt{5000^2 - (y - 5000)^2}$, as seen in Fig. 17. The feasible region for this problem is $\{(x, y): d(x, y) \leq d_{max}, x < -5{,}000\}$, with $d_{max} = 900, 800, 700, 600$ m. Since in this case the water depth gradually gets smaller as we approach the shoreline on the right side, it is interesting to investigate what happens if the depth constraints are varied.

As seen in Fig. 17, not all sensors go to their respective depth constraint. This is expected because the bottom topography is not flat. As predicted, the sensors obtained with varying depth constraints lie on the bisector of the semicircle. As the water depth increases, the location of the sensors move away from the shore and closer to the subduction zones.

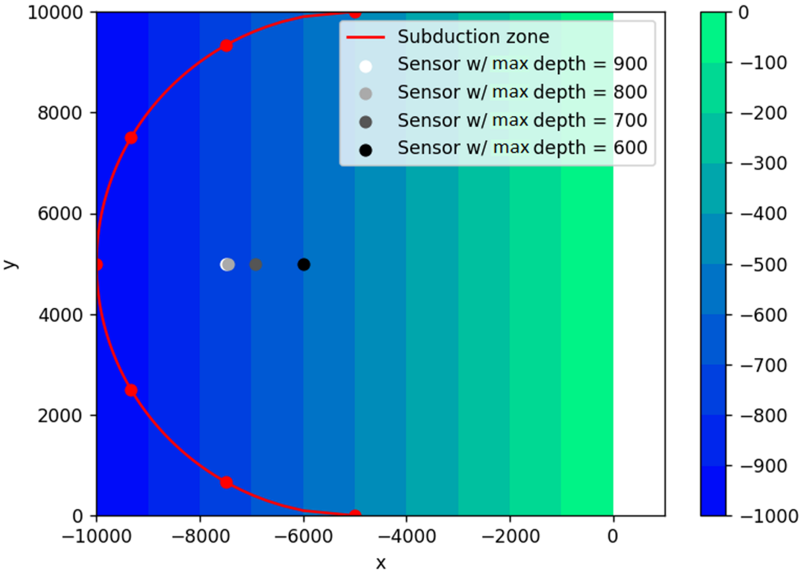

Figure 17 Optimal location of one detection sensor on a sloping bottom with varying depth constraints. Full-size DOI: 10.7717/peerj-cs.685/fig-17

However, the sensor cannot get too close to the semicircle to guarantee detection time from all the possible sources. This can be seen on the obtained locations of sensors for depths 800 and 900, *i.e.*, the two locations are almost identical. This is possible because the feasible region for d_{max} = 800 m is contained in the feasible region for d_{max} = 900 m.

Multiple detection sensors

Step bottom

Using the same setup as the previous one, the PSO algorithm was run to solve the optimal location of two detection sensors. As seen in Fig. 18, the obtained locations of the sensors are on the boundary of the feasible region. This is expected because the sensors must be placed as close as possible to the source points and at the same time must satisfy the depth constraint. It also makes sense that neither sensor was located right in the middle, since one sensor can account for waves coming from the upper side while the other can account for waves coming from the lower side.

Sloping bottom

In this example, the optimal locations of sensors are more spread out to account for waves coming from different angles. The results are illustrated in Fig. 19. Obviously, increasing the number of sensors will always decrease the wave detection time. As shown in Fig. 20, using two sensors instead of one allows the waves to be detected about 37 s earlier or 68% faster. However, using three or more sensors would be less efficient, as this only decreases the detection time by 1 s or even less.

Sloping bottom with an island

This simulation starts with the same setup as the sloping bottom simulation, only this time we introduce a small island in the middle of the domain. In this case, we expect to see

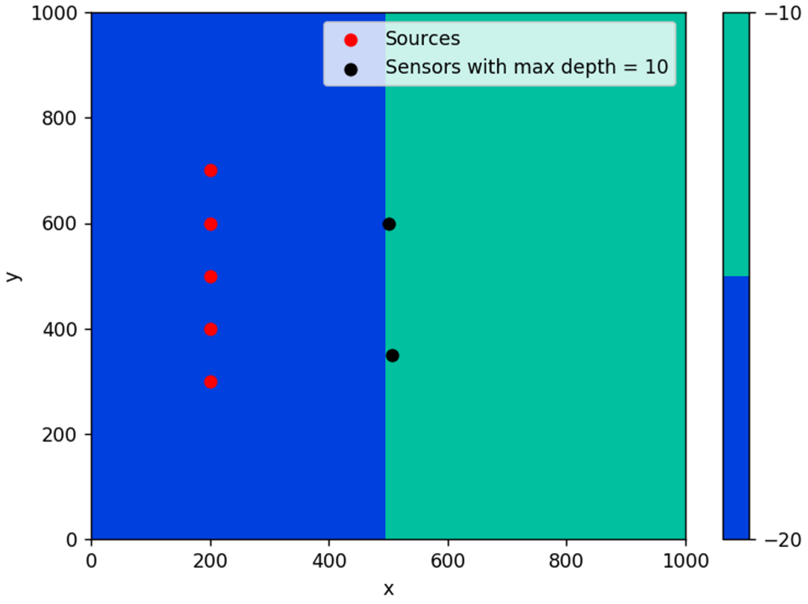

Figure 18 **Optimal location of two detection sensors on a step bottom.**
Full-size DOI: 10.7717/peerj-cs.685/fig-18

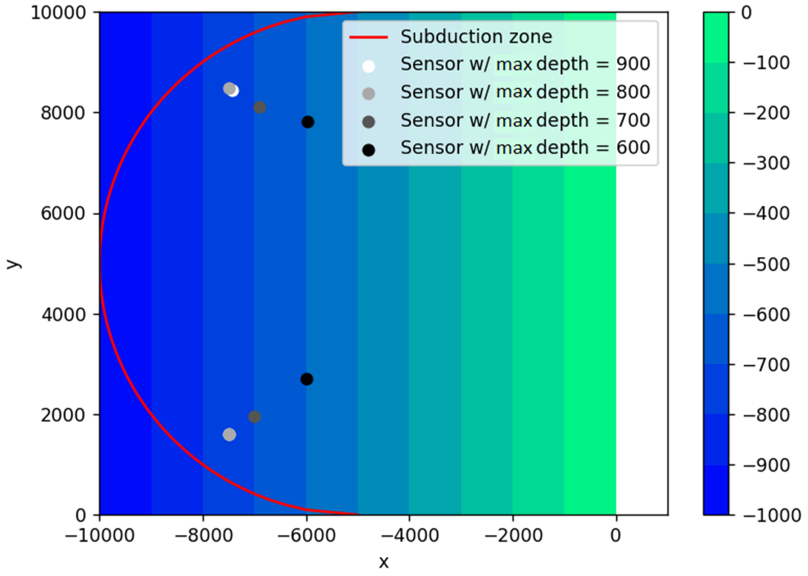

Figure 19 **Optimal location of two detection sensors on a sloping bottom with varying depth constraints.**
Full-size DOI: 10.7717/peerj-cs.685/fig-19

interesting results on small depth constraints since there are two “shallow” areas here: the center island and the coastline on the right. It can be seen in Fig. 21 that sensors with depth constraints of 500, 700 m are not really affected by the island. However for maximum depth of 300 and 100 m, the pair of sensors are spread apart, with one approaching the island and the other getting closer to the coastline on the right. Because the domain is symmetric with respect to the line y = 5,000 m, two solutions are obtained. This is illustrated in Fig. 21. This configuration could not be tackled in *Ferrolino et al. (2020)*

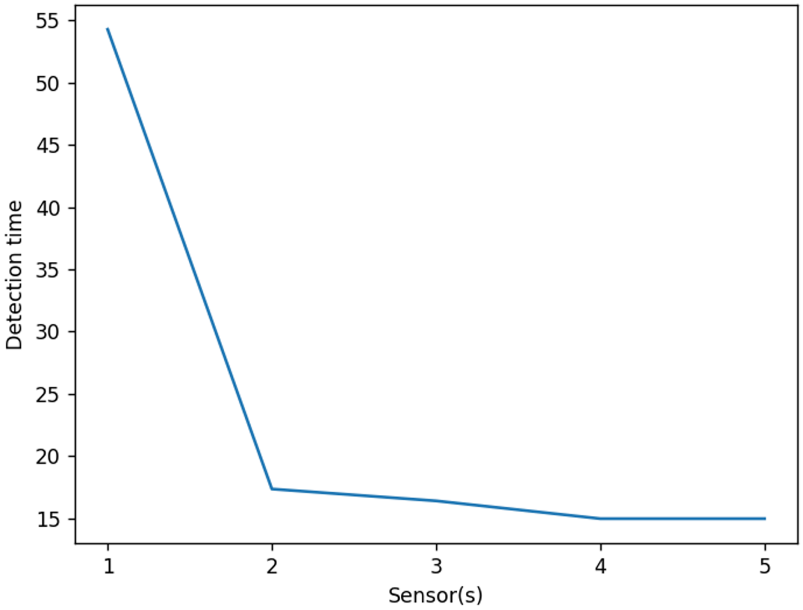

Figure 20 **Effects of placing multiple detection sensors on wave detection time (d_{max} = 700 m).**
Full-size DOI: 10.7717/peerj-cs.685/fig-20

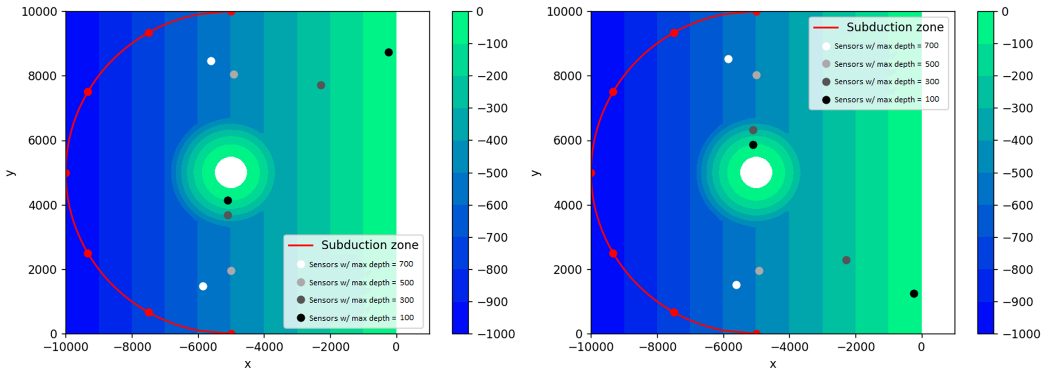

Figure 21 **Two obtained solutions for the optimal placement of two sensors on asloping bottom with an island subject to varying depth constraints.** Full-size DOI: 10.7717/peerj-cs.685/fig-21

because the presence of an island in the simulation requires the use of the nonlinear SWE with wet-dry procedure.

The simulations we have presented so far were formulated so that the optimal solutions can be verified based on the geometry of the subduction zone and the domain. The obtained results using PSO yield solutions that agree with the expected geometric solutions. These tests were done to check that the method works. In a real-world scenario, the optimal placement of sensors is not trivial because the subduction zones do not necessarily follow a geometric pattern. Additionally, the bottom topography of oceans is uneven, making the computation of travel time more complicated. Because we use the 2D nonlinear SWE in simulating tsunami waves, our proposed method can handle complex bathymetric profiles and arbitrary tsunami sources. In our following and last

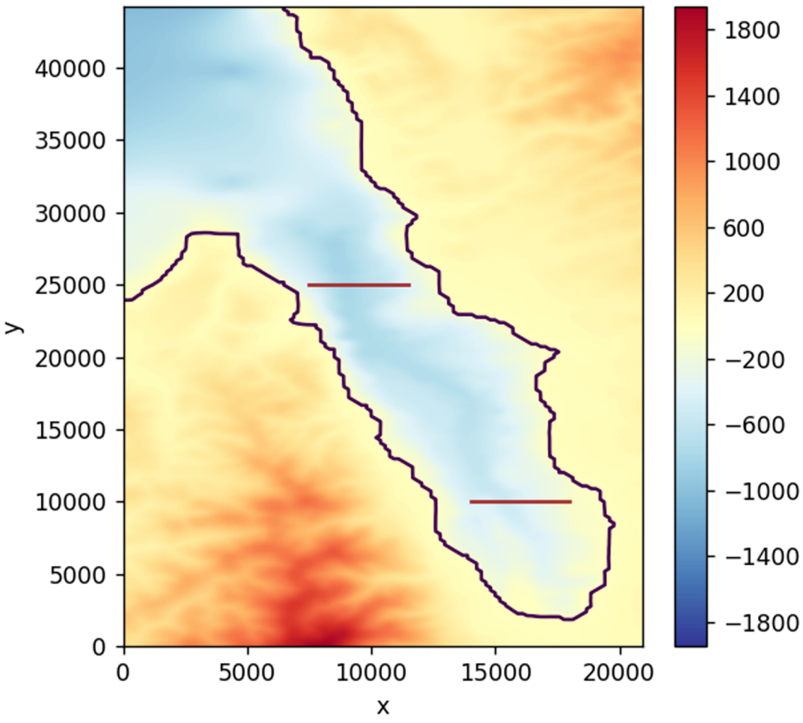

Figure 22 Filled contour plot of Palu Bay. The possible sources of the tsunami are landslides along the lines (in red) as proposed by *Heidarzadeh, Muhari & Wijanarto (2018)*.
Full-size DOI: 10.7717/peerj-cs.685/fig-22

example, we apply our tsunami sensor detection model to one of the more recent tsunami incidents: the 2018 Palu tsunami.

Application on real events: 2018 Palu tsunami

On 28 September 2018, Central Sulawesi was struck by a 7.5-Richter-scale earthquake, which was followed by a tsunami sweeping through the city of Palu.

To simulate the tsunami, we use a gridded bathymetry of Palu Bay provided by BATNAS Indonesia, with a grid size of 185 m × 185 m. Landslides are generated from two subduction zones that were proposed by *Heidarzadeh, Muhari & Wijanarto (2018)* using backward tsunami ray tracing, see Fig. 22.

Solving the optimization problem using PSO on this simulation without constraints found the optimal sensor placement at depth $d = 595.48$ m with coordinates: longitude = 119.81583, latitude = −0.74583. As can be seen in Fig. 23, this location is approximately in the middle of the region bounded by the two subduction zones. Adding some more depth constraints, from 450 to 150 m, consistently moved the sensor closer to the shore.

Assuming no constraints, placing the detection sensor according to this simulation would allow the waves coming from either side to be detected in just about 93 s. This shaves off up to 240 s compared to the case when the sensor was placed more to the south, closer to Talise Beach, which would then take either 113 s or 333 s depending on the source. If a depth constraint of 150 m is applied, the optimally located sensor would still be able to detect the tsunami in about 106 s, saving up to 227 s.

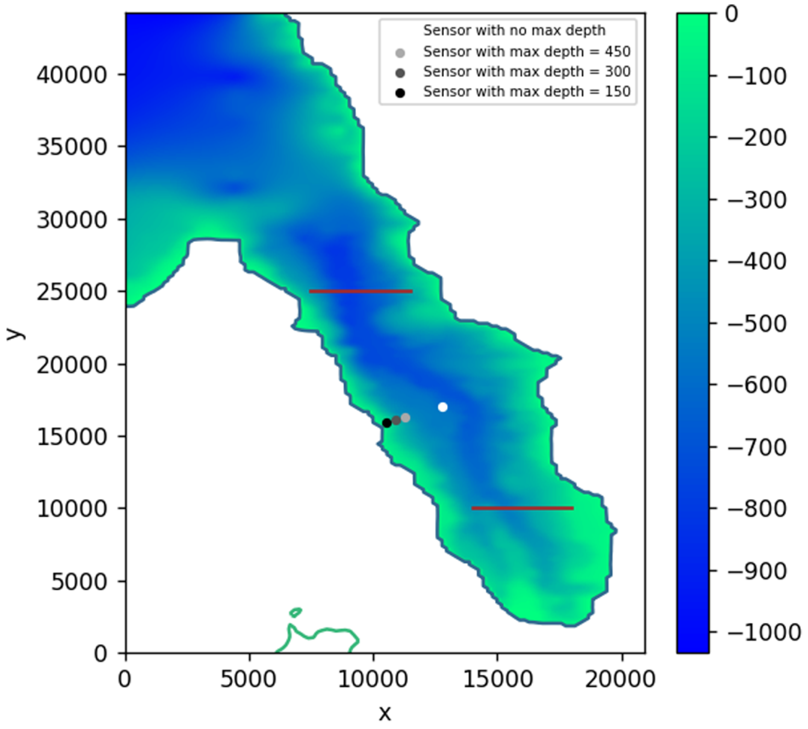

Figure 23 **Optimal location of one detection sensor, with varying depth constraints, based on 2018 Palu tsunami incident.** Full-size DOI: 10.7717/peerj-cs.685/fig-23

CONCLUSION

To conclude, this paper has accomplished two significant results. First, we have derived a robust and accurate numerical shallow water model (SWE) for simulating water waves in different situations. The wet-dry procedure that are included in the calculation can simulate wave propagates over a dry area. Secondly, we have integrated this shallow water model in conjunction with the meta-heuristic PSO algorithm to solve various optimization problems of placing tsunami detection sensors, including the 2018 Palu tsunami incident. The latter findings proved that strategic placement of detection sensors for tsunami warning systems can drastically improve detection time, giving the responsible parties more time to evacuate the citizens.

In the previous work by *Ferrolino et al., 2020*, the tsunami sensors can be placed anywhere in the water domain. Their results yield the placement of sensors near the subduction zones for faster detection time. However, subduction zones are usually located in the deep ocean, making the placement of sensors costly. In the Philippines, tsunami warning systems are installed near the shores. Because of this depth constraint, we modify the optimization problem proposed in *Ferrolino et al. (2020)*. We use a penalty method to solve the constrained minimization problem, which is a generalization of the tsunami detection problem presented in *Ferrolino et al. (2020)*. The arising minimization problem is solved using PSO, a robust and easy-to-implement global optimization algorithm.

The benchmark tests for the tsunami sensors location problem are formulated so that the optimal solution can be inferred geometrically. This way, we can assess if the obtained solution by PSO is correct. Our numerical results agree with the expected solution. However, it is still interesting to know what happens if other numerical optimization algorithms are used. A comparative analysis of recent metaheuristic optimization algorithms applied to tsunami sensor detection problems merits a separate study.

In our test simulations, the possible tsunami sources are evenly placed along the subduction zone. However, in reality, some places in the subduction zone may be more tectonically active than others. This is a limitation of the current study. For future research, one can use time-series data of earthquakes within the subduction zone and use a clustering algorithm to identify coordinates along the zone that will most likely be a tsunami source.

ADDITIONAL INFORMATION AND DECLARATIONS

Funding

This work was supported by Institut Teknologi Bandung. The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Grant Disclosures

The following grant information was disclosed by the authors:
Institut Teknologi Bandung.

Competing Interests

The authors declare that they have no competing interests.

Author Contributions

- Ikha Magdalena conceived and designed the experiments, performed the experiments, analyzed the data, performed the computation work, prepared figures and/or tables, authored or reviewed drafts of the paper, and approved the final draft.
- Raynaldi La'lang conceived and designed the experiments, performed the experiments, analyzed the data, performed the computation work, prepared figures and/or tables, authored or reviewed drafts of the paper, and approved the final draft.
- Renier Mendoza conceived and designed the experiments, performed the experiments, analyzed the data, performed the computation work, prepared figures and/or tables, authored or reviewed drafts of the paper, and approved the final draft.
- Jose Ernie Lope analyzed the data, authored or reviewed drafts of the paper, and approved the final draft.

Data Availability

The following information was supplied regarding data availability:

The data used in benchmark tests are available at the NOAA website:

1. Conical Island https://nctr.pmel.noaa.gov/benchmark/Laboratory/Laboratory_ConicalIsland/index.html

2. Monai Valley https://nctr.pmel.noaa.gov/benchmark/Laboratory/Laboratory_MonaiValley/

The bathymetry data for Palu simulation are available in http://batnas.big.go.id/pencarian?kategori=Batimetri&keyword=palu/

The Python code used to run the simulations are available at GitHub: https://github.com/raynaldilalang/SWE-PSO.

REFERENCES

Astrakova AS, Bannikov DV, Cherny SG, Lavrentiev MM Jr. 2009. The determination of the optimal sensors' location using genetic algorithm. In: *Proceedings of the 3rd Nordic EWM Summer School*. 5–22.

Behrens J. 2010. Numerical methods in support of advanced tsunami early warning. In: *Handbook of Geomathematics*. Berlin Heidelberg: Springer, 601–621.

Bernard EN, Meinig C. 2011. History and future of deep-ocean tsunami measurements. In: *OCEANS'11 MTS/IEEE KONA*. Piscataway: IEEE, 1–7.

Briganti R, Dodd N. 2009. Shoreline motion in nonlinear shallow water coastal models. *Coastal Engineering* **56(5–6)**:495–505 DOI 10.1016/j.coastaleng.2008.10.008.

Briggs MJ, Synolakis CE, Harkins GS, Green DR. 1995. Laboratory experiments of tsunami runup on a circular island. *Pure and Applied Geophysics PAGEOPH* **144(3–4)**:569–593 DOI 10.1007/BF00874384.

Brocchini M, Dodd N. 2008. Nonlinear shallow water equation modeling for coastal engineering. *Journal of Waterway, Port, Coastal, and Ocean Engineering* **134(2)**:104–120 DOI 10.1061/(ASCE)0733-950X(2008)134:2(104).

Catalan PA, Gubler A, Cañas J, Zuñiga C, Zelaya C, Pizarro L, Valdes C, Mena R, Toledo E, Cienfuegos R. 2020. Design and operational implementation of the integrated tsunami forecast and warning system in Chile (SIPAT). *Coastal Engineering Journal* **62(3)**:373–388 DOI 10.1080/21664250.2020.1727402.

Cecioni C, Bellotti G, Romano A, Abdolali A, Sammarco P, Franco L. 2014. Tsunami early warning system based on real-time measurements of hydro-acoustic waves. *Procedia Engineering* **70(6)**:311–320 DOI 10.1016/j.proeng.2014.02.035.

Chanson H. 2008. A simple solution of the laminar dam break wave. *Journal of Applied Fluid Mechanics* **1(1)**:1–10.

Chao W, Wu T, Ma K, Kuo Y, Wu Y, Zhao L, Chung M, Wu H, Tsai Y. 2018. The large Greenland landslide of 2017: was a tsunami warning possible? *Seismological Research Letters* **89(4)**:1335–1344 DOI 10.1785/0220170160.

Doi K. 2003. Tsunami warning system in japan. In: *Early Warning Systems for Natural Disaster Reduction*. Berlin: Springer, 537–541.

DOST, DOST-PHILVOLCS, DOST-ASTI. 2021. Tsunami early warning system. *Available at http://tews.asti.dost.gov.ph/*.

E2S Warning Signals. 2021. Local tsunami early warning system for high risk coastal communities of the Philippines. *Available at https://www.e2s.com/case-studies/latest*.

Ferrolino A, Mendoza R, Magdalena I, Lope JE. 2020. Application of particle swarm optimization in optimal placement of tsunami sensors. *PeerJ Computer Science* **6(1)**:e333 DOI 10.7717/peerj-cs.333.

Ferrolino AR, Lope JEC, Mendoza RG. 2020. Optimal location of sensors for early detection of tsunami waves. In: *Computational Science–ICCS 2020*. Cham: Springer International Publishing, 562–575.

Geyer A, Quirchmayr R. 2017. Shallow water equations for equatorial tsunami waves. *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences* **376(2111)**:100.

Grilli S, Tappin D, Carey S, Watt S, Ward S, Grilli A, Engwell S, Zhang C, Kirby J, Schambach L, Muin M. 2019. Modelling of the tsunami from the December 22, 2018 lateral collapse of Anak Krakatau volcano in the Sunda Straits, Indonesia. *Scientific reports* **9(1)**:669 DOI 10.1038/s41598-019-48327-6.

Heidarzadeh M, Muhari A, Wijanarto AB. 2018. Insights on the source of the 28 September 2018 Sulawesi tsunami, Indonesia based on spectral analyses and numerical simulations. *Pure and Applied Geophysics* **176(1)**:25–43 DOI 10.1007/s00024-018-2065-9.

Heidarzadeh M, Satake K. 2017. Possible dual earthquake–landslide source of the 13 November 2016 Kaikoura, New Zealand tsunami. *Pure and Applied Geophysics* **174**:3737–3749 DOI 10.1007/s00024-017-1637-4.

Joseph A. 2011. *Tsunamis: detection, monitoring, and early-warning technologies*. First Edition. Cambridge: Academic Press.

Kennedy J, Eberhart R. 1995. Particle swarm optimization. *Proceedings of ICNN'95-International Conference on Neural Networks* **4**:1942–1948 vol.4 DOI 10.1109/ICNN.1995.488968.

Kennedy J, Eberhart R. 1995. Particle swarm optimization. *Proceedings of ICNN'95-International Conference on Neural Networks* **4**:1942–1948 DOI 10.1109/ICNN.1995.488968.

Le Roux D, Staniforth A, Lin C. 1998. Finite elements for shallow-water equation ocean models. *Monthly Weather Review* **126(7)**:20–29 DOI 10.1175/1520-0493(1998)126<1931:FEFSWE>2.0.CO;2.

Lee E, Jung T, Shin S. 2020a. Numerical and probabilistic study on the optimal region for tsunami detection instrument deployment in the Eastern Sea of Korea. *Applied Sciences* **10(17)**:6071 DOI 10.3390/app10176071.

Lee E, Jung T-H, Shin S. 2020b. A study on optimal deployment of tsunami observation instrument in Korea. *Coastal Engineering Proceedings* **(36v)**:24 DOI 10.9753/icce.v36v.currents.24.

Liu PL-F, Yeh H, Synolakis C. 2008. Advanced numerical models for simulating tsunami waves and runup. In: *Advances in Coastal and Ocean Engineering*. Singapore: World Scientific.

Lohan P, Dube P, Agrawal M. 2020. A novel 3d topology design for underwater sensor networks. In: *Global Oceans 2020: U.S. Gulf Coast*. Singapore: IEEE.

Magdalena I, Erwina N, Pudjaprasetya SR. 2015. Staggered momentum conservative scheme for radial dam break simulation. *Journal of Scientific Computing* **65(3)**:867–874 DOI 10.1007/s10915-015-9987-5.

Magdalena I, Rif'atin HQ, Reeve DE. 2020. Seiches and harbour oscillations in a porous semi-closed basin. *Applied Mathematics and Computation* **369(1–2)**:124835 DOI 10.1016/j.amc.2019.124835.

Meza J, Catalán PA, Tsushima H. 2019. A multiple-parameter methodology for placement of tsunami sensor networks. *Pure and Applied Geophysics* **177(3)**:1451–1470 DOI 10.1007/s00024-019-02381-3.

Mezura-Montes E, Coello CAC. 2011. Constraint-handling in nature-inspired numerical optimization: past, present and future. *Swarm and Evolutionary Computation* **1(4)**:173–194 DOI 10.1016/j.swevo.2011.10.001.

Mulia IE, Hirobe T, Inazu D, Endoh T, Niwa Y, Gusman AR, Tatehata H, Waseda T, Hibiya T. 2020. Advanced tsunami detection and forecasting by radar on unconventional airborne observing platforms. *Scientific Reports* **10(1)**:25 DOI 10.1038/s41598-020-59239-1.

Mulia IE, Satake K. 2020. Developments of tsunami observing systems in japan. *Frontiers in Earth Science* **8**:25 DOI 10.3389/feart.2020.00145.

Mungov G, Eble M, Bouchard R. 2012. Dart tsunameter retrospective and real-time data: a reflection on 10 years of processing in support of tsunami research and operations. *Pure and Applied Geophysics* **170(9–10)**:1369–1384 DOI 10.1007/s00024-012-0477-5.

Murjaya J. 2012. *Tsunami early warning service guidebook for InaTEWS*. Indonesia: BMKG.

Navarrete P, Cienfuegos R, Satake K, Wang Y, Urrutia A, Benavente R, Catala´n PA, Crempien J, Mulia I. 2020. Sea surface network optimization for tsunami forecasting in the near field: application to the 2015 Illapel earthquake. *Geophysical Journal International* **221(3)**:1640–1650 DOI 10.1093/gji/ggaa098.

Okal EA, Synolakis CE. 2015. Sequencing of tsunami waves: why the first wave is not always the largest. *Geophysical Journal International* **204(2)**:719–735 DOI 10.1093/gji/ggv457.

Paros J, Bernard E, Delaney J, Meinig C, Spillane M, Migliacio P, Tang L, Chadwick W, Schaad T, Stalin S. 2011. Breakthrough underwater technology holds promise for improved local tsunami warnings. In: *2011 IEEE Symposium on Underwater Technology and Workshop on Scientific Use of Submarine Cables and Related Technologies*. Piscataway: IEEE, 1–9.

Pudjaprasetya SR, Magdalena I. 2014. Momentum conservative schemes for shallow water flows. *East Asian Journal on Applied Mathematics* **4(2)**:152–165 DOI 10.4208/eajam.290913.170314a.

Satake K. 2014. Advances in earthquake and tsunami sciences and disaster risk reduction since the 2004 Indian Ocean tsunami. *Geoscience Letters* **1(15)**:1–13 DOI 10.1186/2196-4092-1-1.

Shi Y, Eberhart R. 1998. A modified particle swarm optimizer. In: *Proceedings of IEEE International Conference on Evolutionary Computation*. Piscataway: IEEE, 69–73.

Synolakis C. 1987. The runup of solitary waves. *Journal of Fluid Mechanics* **185**:523–545 DOI 10.1017/S002211208700329X.

Thacker WC. 1981. Some exact solutions to the nonlinear shallow-water wave equations. *Journal of Fluid Mechanics* **107(-1)**:499–508 DOI 10.1017/S0022112081001882.

Triantafyllou I, Gogou M, Mavroulis S, Lekkas E, Papadopoulos GA, Thravalos M. 2021. The tsunami caused by the 30 October 2020 Samos (Aegean Sea) Mw7.0 earthquake: hydrodynamic features, source properties and impact assessment from post-event field survey and video records. *Journal of Marine Science and Engineering* **9(1)**:68.

UN-ESCAP. 2009. Tsunami early warning systems in the Indian Ocean and Southeast Asia: report on regional unmet needs. New York: United Nations.

Wagih MM, Elkamchouchi HM, Lazinica A. 2009. Application of particle swarm optimization algorithm in smart antenna array systems. In: Lazinica A, ed. *Particle Swarm Optimization*. London: InTech, 461–476 DOI 10.5772/6765.

Wang J-F, Li L-F. 2008. Improving tsunami warning systems with remote sensing and geographical information system input. *Risk Analysis* **28(6)**:1653–1668 DOI 10.1111/j.1539-6924.2008.01112.x.

Wu K, Chen P, Ghattas O. 2021. A fast and scalable computational framework for goal-oriented linear bayesian optimal experimental design: application to optimal sensor placement. ArXiv preprint. *Available at https://arxiv.org/abs/2102.06627*.

Yang X-S. 2014. *Nature-inspired optimization algorithms*. Amsterdam, Netherlands: Elsevier.

Zergani S, Aziz ZA, Viswanathan KK. 2015. A shallow water model for the propagation of tsunami via Lattice Boltzmann method. *IOP Conference Series: Earth and Environmental Science* **23**:12007 DOI 10.1088/1755-1315/23/1/012007.

Zhou C, Gao L, Gao H-B, Zan K. 2006. Particle swarm optimization for simultaneous optimization of design and machining tolerances. In: Wang TD, Li X, Chen SH, Wang X, Abbass H, Iba H, Chen GL, Yao X, eds. *Simulated Evolution and Learning. SEAL 2006. Lecture Notes in Computer Science*. Vol. 4247. Berlin, Heidelberg: Springer.