

Optimal Location of Sensors for Early Detection of Tsunami Waves

Angelie R. Ferrolino, Jose Ernie C. Lope, and Renier G. Mendoza

Institute of Mathematics, University of the Philippines Diliman
arferrolino@up.edu.ph, ernie@math.upd.edu.ph, rmendoza@math.upd.edu.ph

Abstract. Tsunami early detection systems are of great importance as they provide time to prepare for a tsunami and mitigate its impact. In this paper, we propose a method to determine the optimal location of a given number of sensors to report a tsunami as early as possible. The rainfall optimization algorithm, a population-based algorithm, was used to solve the resulting optimization problem. Computation of wave travel times was done by illustrating the kinematics of a wave front using a linear approximation of the shallow water equations.

Keywords: shallow water equations · sensor location · tsunami early warnings · rainfall optimization algorithm

1 Introduction

Tsunamis are considered as one of the most powerful and destructive natural disasters. They are a series of waves caused by a rapid and massive displacement of the seafloor or disruption of standing water. Earthquakes, volcanic eruptions, underwater landslides and meteor impacts all have the potential of generating a tsunami. However, the most common cause is undersea earthquakes, generated in subduction zones. The water above such event is disturbed significantly by the uplifting, that it create waves travelling at around 500 miles per hour. Moreover, their wavelength is much longer than normal sea waves, so they build up to higher heights as the depth of water decreases. With the large volume of water moving at high speed, tsunamis can cause massive destruction to the physical environment and loss of human lives.

The Philippines is one of the most disaster-prone countries due to its geographic location. As such, it has a high risk of exposure to intense tropical cyclones, seismic activities and tsunamis [24]. Up to date, a total of 41 tsunamis occurred since 1589 [5]. Thus, compared to other countries, occurrence of tsunami in the Philippines is more than average, but still moderate. However, as stated in [13], tsunami research in the Philippines received less than its deserved attention. Moreover, lack of awareness and sufficient preparation may further increase the hazards. So, developing techniques to assess risks and to mitigate impacts of a tsunami are of great importance. There are a number of methods focused in reduction of tsunami hazards and risk management in literature [9, 20, 10]. In

this study, we aim to provide early tsunami warnings by determining the optimal location of tsunami sensors that can guarantee minimal possible tsunami detection time.

The traditional instrument used for detecting a tsunami is called a tide gauge, which is usually located near the coast. It measures changes in the sea level relative to some height reference. While the tide gauge is capable of detecting a tsunami, they are at a worst location, since this is where the tsunami is most energetic. Moreover, as they are near the coast, it may take a lot of time for tsunami detection, leaving only a small amount of time for evacuation. Hence, we consider using deep-ocean sensors, which are capable of reporting a tsunami in the open ocean. Thus, this technology provides a relatively secure and rapid detection of tsunami, but at a cost. Details regarding these tsunami observing systems were well documented and reviewed in [22, 18]. In particular, we will consider the bottom pressure recorders (BPRs) as our tsunami sensor. These sensors measure changes in water pressure and seismic activity. They use an acoustic link to transmit data on surface buoys, which are then relayed via satellite to ground stations [8]. For optimal sensor placement, we assume that the tsunami originates from a fixed location, e.g., subduction zones. Rogue waves [7] are not considered in this work because of the uncertainty of their origin.

Studies regarding selection of location of tsunami sensors for tsunami warnings are limited. Some were based on expert judgments while considering various deciding factors like financial limitations [3], and legal aspects such as geographical boundaries or legal jurisdictions [1]. There are also researches that incorporated accurate estimation of tsunami parameters [17, 16]. An attempt to encompass tsunami warning efficacy has been proposed in [19, 23]. They identified location of tsunami sensors based on several criteria (e.g. installation conditions), though no optimization algorithm was applied. Bradock et al. [6] considered six potential buoy sites, and they determined the optimal location of a minimum number of BPRs that will maximize the population being warned. However, they considered a fixed average speed of wave travel, and thus, bathymetric data cannot be taken into account. Astrakova et al. [4] developed an optimal sensor location problem, which guarantees minimal possible time of tsunami detection. However, the method used in computing the wave travel time is very slow, and takes too much amount of computer memory.

The objective of this paper is to improve the optimal sensor location problem introduced by Astrakova. We illustrate kinematics of a wavefront, using an approximation of wave velocity from the linear shallow water equations, which results to a faster calculation of wave travel time and low computer cost. Finally, we test our method to a simple problem, then apply it to a real world problem of optimal sensor location in one of the areas in the Philippines, near a subduction zone.

To solve the resulting optimization problem, we use a population-based algorithm called the Rainfall Optimization Algorithm (RFO) introduced by Kaboli et al. [11]. RFO is a nature-inspired algorithm which is modelled on raindrop flow over a mountainous surface and has been shown to be effective, fast and ca-

pable of solving various optimization problems. It has been tested on benchmark functions against the genetic algorithm (GA), the particle swarm optimization algorithm (PSO) and the group search optimizer (GSO), and it outperforms them on most functions, as it provides better solutions and faster convergence.

The rest of the paper is organized as follows: in the next section, we discuss the methods used in solving the optimal sensor location problem. Results of the optimal location of tsunami sensors in different domains and bathymetries are presented in the third section. Finally, we summarize our results and provide recommendations for future work.

2 Methodology

In this section, we discuss the optimization problem, the method for computing the wave travel time, and details of the rainfall optimization algorithm.

2.1 The Optimization Problem

Let Ω be the domain with parts of water, $\mathbf{D} \in \Omega$ be a part of the domain where tsunami sensors can be placed, and \mathbf{P} be the subduction zone. For an illustration, please refer to Figure 1. The problem of interest is to find the location of L sensors such that any seismic event on \mathbf{P} will be detected after minimal possible time. Let $\{p_j\}_{j=1}^P$ denote the points in the subduction zone, where $p_j = (x_j, y_j) \in \mathbf{P}$, and $q_i = (x_i, y_i) \in \mathbf{D}$ ($i = 1, \dots, L$) be the coordinates of the L sensors. Also, we denote by $\mathbf{Q} = \{q_1, \dots, q_L\}$ the set of L sensors representing a possible solution to the optimization problem.

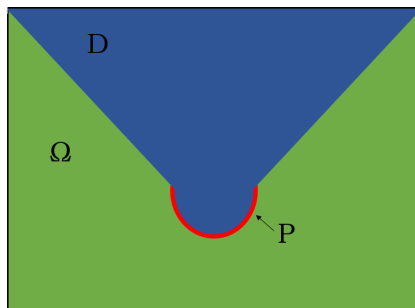


Fig. 1. Illustration of a Domain

Suppose that a disturbance arises on a source $p_j \in \mathbf{P}$. This disturbance will propagate over water at a certain speed. We are interested in finding the minimal time it takes for the disturbance on the source p_j to arrive at some water point

$x \in \mathbf{D}$. Let γ be one of the ways connecting p_j and x , and τ_γ be its travel time. We denote $\tau(p_j, x)$ be the minimal time to approach x from p_j , i.e.,

$$\tau(p_j, x) = \min_{\gamma} \tau_\gamma \quad (1)$$

The time required for determination of tsunami wave by \mathbf{Q} is given by

$$t(p_j, \mathbf{Q}) = \min_{1 \leq i \leq L} \tau(p_j, q_i).$$

For guaranteed time registration for any point $p_j \in \mathbf{P}$ by \mathbf{Q} , we set

$$T(\mathbf{Q}) = \max_{1 \leq j \leq P} t(p_j, \mathbf{Q}).$$

Thus the optimization problem is stated as: Find $\mathbf{Q} = \{q_1, \dots, q_L\}$ which gives the minimal value of the function $T(\mathbf{Q})$, i.e.,

$$\min T(\mathbf{Q}), \quad (2)$$

where the number L is given and subject to the phase restriction

$$\mathbf{Q} \in \mathbf{D}.$$

This minimization problem is based on [4].

2.2 Kinematics of a Wave Front

We discuss here how we can illustrate the kinematics of a wave front, which we will use to calculate the wave travel time in (1). We consider the linear approximation of the shallow water equations [21]. In this case, the wave velocity is proportional to the square root of the water depth h :

$$v \approx \sqrt{gh}, \quad h \geq 0,$$

where $g = 9.8 \text{ m/s}^2$ is the gravitational constant. Also, we note that for any propagation velocity distribution in a medium, all the points on a wave front are moving in the orthogonal direction to the frontal line [15]. This will be the basis for the numerical method of the step-by-step wave frontal line advancement that will now be described.

Consider a closed and convex curve (e.g., a circle) as the initial wave front. This wave front is represented by a limited number of computational points (x_i, y_i) ($i = 1, \dots, N$) along the curve. Figure 2(a) presents an example of an initial wave front. Our next aim is to calculate the moving direction for all wave front points to establish their next positions. In this method, the moving direction of the point (x_i, y_i) is determined by the outer-normal of the circle passing through three computational points: (x_{i-1}, y_{i-1}) , (x_i, y_i) and (x_{i+1}, y_{i+1}) . Moving the point (x_i, y_i) in this direction with a distance of $v \cdot \Delta t$, where Δt is the time step, will give us the next position of (x_i, y_i) . Hence, we can compute

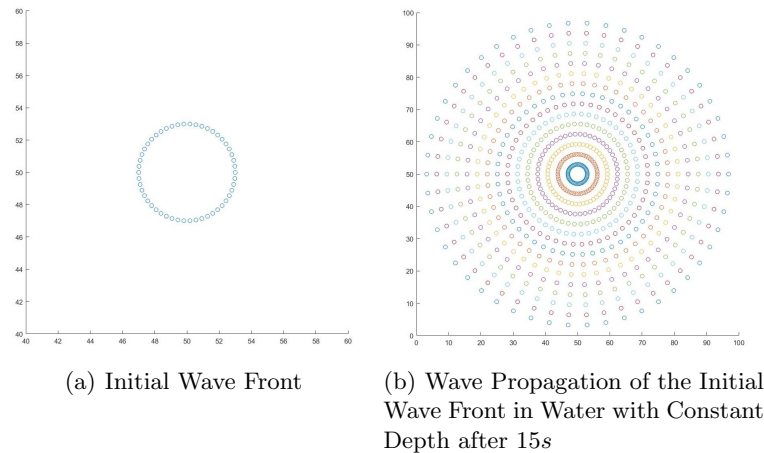


Fig. 2. Kinematics of a Wave Front

the location of all wave front points at the time instant $t = \Delta t$. We repeat this process until we reach the boundaries of the domain. Figure 2(b) demonstrates the wave propagation of the initial wave front in water with constant depth after 15s.

Let (x, y) be the coordinates of a computational point and $z = f(x, y)$ be the time it takes to for the wave to reach the point (x, y) . To determine the travel time, to an arbitrary point (\tilde{x}, \tilde{y}) (not necessarily a computational point) in the domain, we can compute the travel time z of the three nearest computational points, say (x_1, y_1) , (x_2, y_2) and (x_3, y_3) , to (\tilde{x}, \tilde{y}) . Using (x_1, y_1) , (x_2, y_2) , (x_3, y_3) , $f(x_1, y_1)$, $f(x_2, y_2)$ and $f(x_3, y_3)$, we apply linear interpolation to estimate the travel time to (\tilde{x}, \tilde{y}) .

2.3 The Rainfall Optimization Algorithm

In order to solve the optimization problem in (2), we will use a new nature-inspired algorithm called the rainfall optimization algorithm (RFO) [11]. Many optimization problems have been solved using RFO as it was proven to be fast, effective and efficient. Some applications are in solving problems in facility-location [2], economic dispatch [11], and scheduling [12].

RFO is a population-based algorithm which is based on the behavior of raindrops flowing over a mountainous surface. As always, raindrops tend to fall on surfaces with a steeper slope. RFO utilizes this tendency, allowing determination of a solution superior to a guess. Raindrops may also be stuck in puddles (local optimum). However, as raindrops accumulates, it may overflow, allowing drops of water to flow downwards again. RFO simulates this behavior, enabling the algorithm to overcome local optimal solutions and reach the global optimum.

Before presenting the algorithm, we first discuss some important terms. Define

$$D^i := x_{i,1}, x_{i,2}, \dots, x_{i,n} \quad (i = 1, 2, \dots, m)$$

be the raindrop i in a population, where n is the number of variables of optimization variables, m is the number of raindrops in a population and $x_{i,j}$ is the variable of interest in the optimization problem. The generation of raindrops are uniformly randomly distributed and must be within the bounds, if there is any.

Each raindrop randomly produces neighbor points in a neighborhood with a radius vector r . We denote the neighbor points as NP_k^i that satisfies

$$\|(D^i - NP_k^i) \cdot u_j\| \leq \|r \cdot u_j\|, \quad i \in \{1, \dots, m\}, \quad j \in \{1, \dots, n\}, \quad k \in \{1, \dots, np\},$$

where u_j is the unit vector in the j th dimension. Among all the neighbor points, we are interested in finding a dominant neighbor point, i.e., a neighbor point whose cost function value is less than the cost function values of the raindrop and the other neighbor points. We call a drop **active** if it has a dominant neighbor point. Otherwise, the drop is **inactive**.

Algorithm 1 Rainfall Optimization Algorithm

Input: $npop$: population size, np : number of neighbor points on each drop, r : radius vector, Ne : maximum number of explosion process, eb : explosion base, ec : explosion counter, $maxiter$: maximum number of iterations

Output: Location of raindrop with minimal cost function value

1. **Initialization:** Generate randomly the first population of raindrops of size $npop$ such that each raindrop satisfy the constraints. Set $iter = 1$. Set to **active** all the raindrops' status.

2. **Iterative Procedure:**

while $iter \leq maxiter$ and active set of raindrops $\neq \emptyset$ **do**

 Do the following for each active raindrop:

- Generate np neighbor points.
- Obtain the cost function values of drops and their neighbor points.
- If there is a dominant neighbor point, then change the drop's present position to that point. Otherwise, apply explosion process to the drop.
- If there is no dominant neighbor point after Ne times of explosion, set the drop's status to **inactive**.

 Create a merit-order list and remove specific numbers of low-ranking drops or assign a higher Ne to high-ranking drops.

 Set $iter = iter + 1$.

end while

3. **Generation of Minimizer:** Calculate the cost function values for all raindrops. Find the raindrop with the minimum cost function.

If a raindrop has no dominant neighbor point, then it may have already converged the global optimum, or it is stuck in a local optimum (due to insufficient number of neighbor points). To prevent the latter, an explosion process is made. In this process, the raindrop produces new np_{ex} neighbor points where

$$np_{ex} = np \times eb \times ec$$

with eb as the explosion base (indicating the explosion range) and ec as the explosion counter. If the drop still has no dominant neighbor point after doing the explosion process Ne times, we make it inactive.

At the end of the algorithm, we create a merit-order list, which contains the rank of the drops in ascending order. Lower ranking drops may be removed from the population, or higher ranking drops may be given special rights (such as higher number of explosion process). The calculation of rank is given by: $\text{rank}_t^i = \frac{1}{2}\text{order}(C1_t^i) + \frac{1}{2}\text{order}(C2_t^i)$, where

$$(C1_t^i) = F(D^i)|_{\text{at the } t^{\text{th}} \text{ iteration}} - F(D^i)|_{\text{at the } 1^{\text{st}} \text{ iteration}}$$

and

$$(C2_t^i) = F(D^i)|_{\text{at the } t^{\text{th}} \text{ iteration}}$$

Here, F represents the objective function. The step-by-step procedure of the RFO is summarized in Algorithm 1.

3 Results

We first consider a domain with a semicircle subduction zone (in red) as shown in Figure 3.

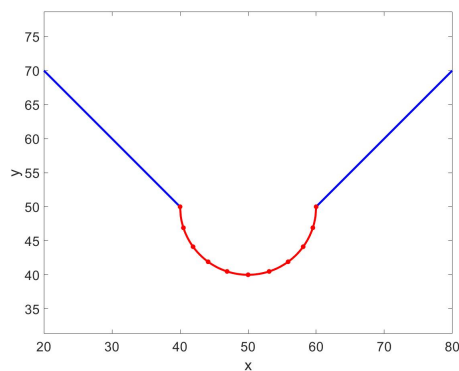


Fig. 3. Domain with a semicircle subduction zone

Assume that we have $L = 2$ sensors, say (x_1, y_1) and (x_2, y_2) . We set x_2 and y_1 to 55 and 45 respectively, while (x_1, y_2) are in the domain $[35, 65] \times [40, 70]$. The plot of time versus x_1 and y_2 , corresponding to the semicircle subduction zone, is presented in Figure 4. From the surface plot, it can be seen how the minimization problem is neither convex nor differentiable. Thus, gradient-based and local search methods might not work. This justifies why a population-based algorithm is an appropriate numerical optimization method to estimate the optimal solution.

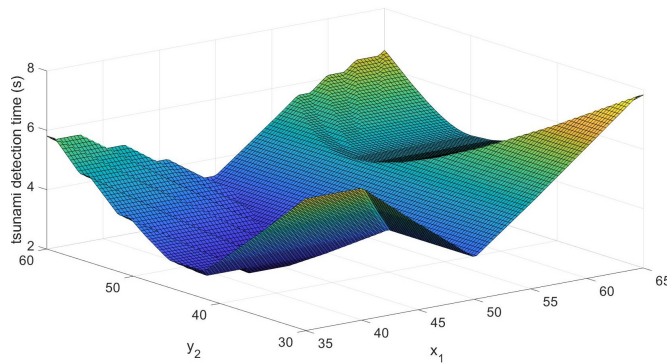


Fig. 4. Surface plot of tsunami detection time versus the coordinates x_1 and y_2

Now, we present some numerical simulations to identify the optimal locations of sensors using various domain profiles. Consider the same domain presented in Figure 3. We let \mathbf{D} be the whole rectangular domain with constant water depth. Figures 5(a) – 5(d) show the optimal location(s) of $L = 1, 2, 3$ and 4 sensor(s), represented by black dots. Here, the red dots in \mathbf{P} are the source points. These results were obtained by taking the average of the locations gathered in 10 independent runs. For the case when $L = 1$, it can be seen that the obtained location is situated at the center of the semicircle. This makes sense geometrically since we wish to minimize the guaranteed time registration from all the source points (red dots) located on the semicircle. One can also observe that as the number of sensors increases, the estimated locations get closer to the subduction zone.

We now study how the number of sensors will affect travel time. Figure 6 shows the plot of the time of tsunami detection versus the number of sensors. We can see here that as the number of sensors increase, the time decreases. Moreover, we can see from Table 3 that there is a significant improvement when we increase the number of sensors from 3 to 4 and little improvement from 4 to 5. Hence, $L = 4$ is a good number of sensors that will give us good tsunami detection time, without the extra cost of additional sensors.

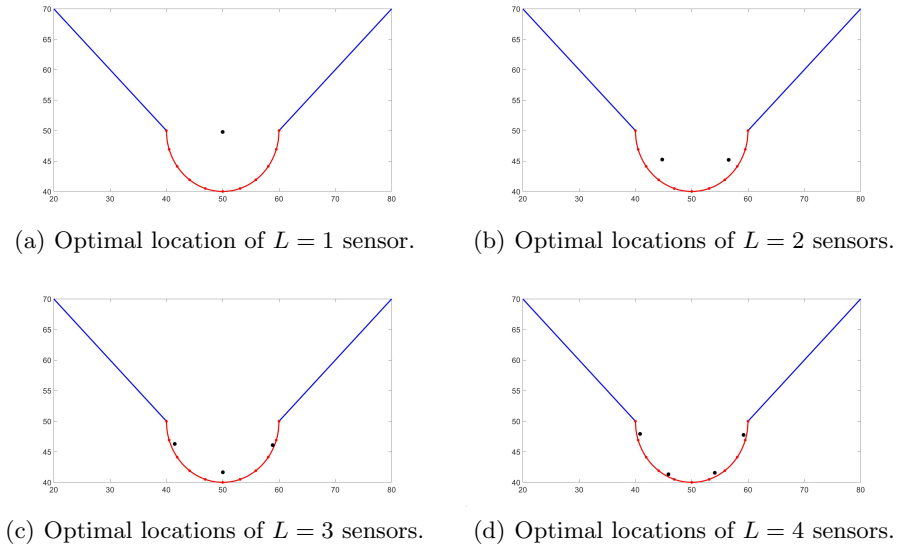


Fig. 5. Numerical results for a semicircle subduction zone.

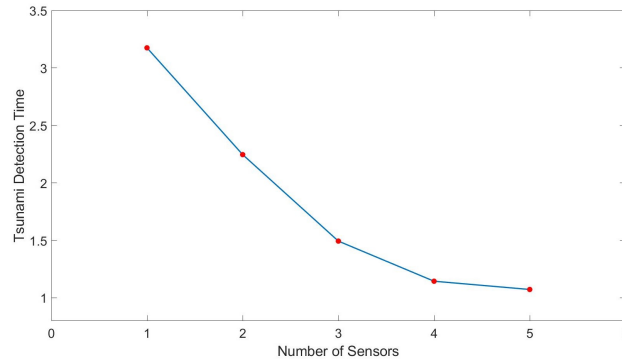


Fig. 6. Effect of increasing the number of sensors on the tsunami detection time.

Number of Sensors (n)	Tsunami Detection Time	Decrease in Time from $n - 1$ to n Sensors
1	3.1756	-
2	2.2465	0.9291
3	1.4940	0.7525
4	1.1449	0.3491
5	1.0731	0.0731

Table 1. Decrease in tsunami detection time as the number of sensors is increased.

Next, we apply our method to a real-world problem of sensor location in the Cotabato Trench. The Cotabato trench is an oceanic trench in the Pacific Ocean, located off the southwestern coast of Mindanao in the Philippines. This trench is one of the main structures around the Philippines likely to be associated with tsunamigenic earthquakes. One example is the tsunami generated by the 1976 Moro Gulf earthquake, which is considered as one of the most devastating disasters in the history of the Philippine islands [14]. Figure 7(a) shows a portion of the Cotabato Trench. We let the subduction zone \mathbf{P} to be the red line, and \mathbf{D} be the water surface above the subduction zone. The corresponding bathymetric profile of this trench is shown in Figure 7(b).

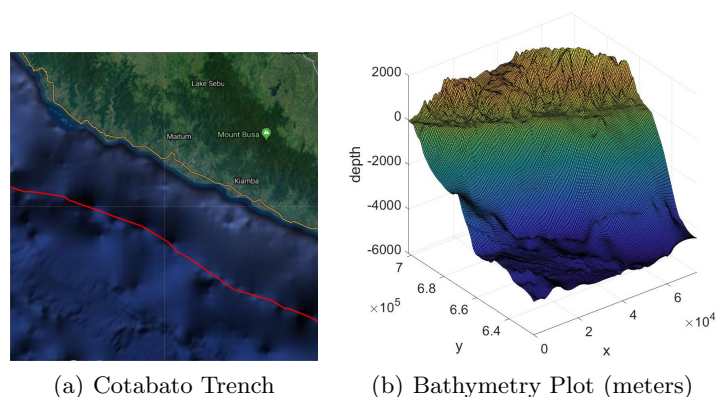


Fig. 7. Profile of the Cotabato Trench.

Figure 8(a), 8(b), 8(c), 8(d), 8(e) and 8(f) present the optimal location of $L = 1, 2, 3, 4, 5$ and 6 sensor/s (blue dots), respectively. These results were obtained by taking the average of the locations gathered in 10 independent runs. The plot of the time of tsunami detection versus the number of sensors is shown in Figure 9. The values of time in dependence to the number of sensors is presented in Table 3. Similar to what we did earlier, we may see from here that $L = 5$ is a good number of sensors for this problem.

4 Conclusion and Recommendations

We considered the problem of optimal tsunami sensors placement for early tsunami warnings. The computation of wave travel times were done by producing the kinematics of a wave front using an approximation of wave velocity derived from the linear shallow water equations. The Rainfall Optimization Algorithm was used to solve the optimization problem. We first applied our model to a simple problem having a semicircle subduction zone for testing. The obtained results for this test problem are geometrically sensible. Then we applied our

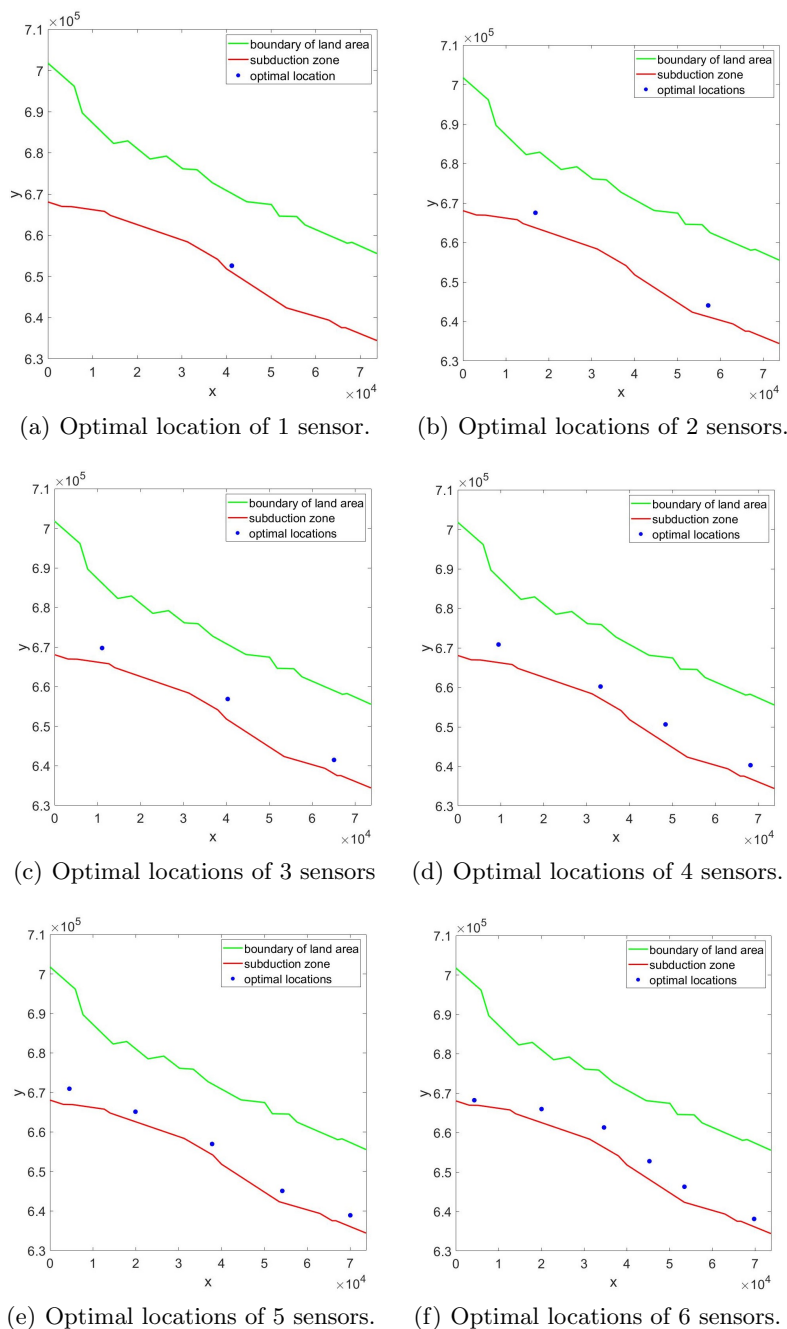


Fig. 8. Numerical results for a portion of the Cotabato Trench.

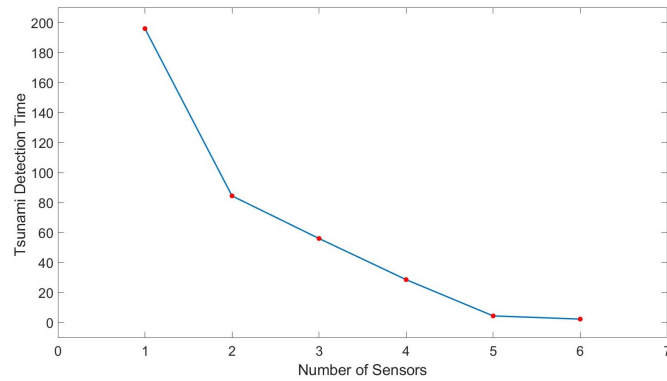


Fig. 9. Number of sensors versus time in the case of Cotabato Trench

Number of Sensors (n)	Tsunami Detection Time	Decrease in Time from $n - 1$ to n Sensors
1	195.9816 s	-
2	84.5089 s	111.4727
3	56.0995 s	28.4094
4	28.6910 s	27.4085
5	4.5239 s	24.1671
6	2.4007 s	2.1232

Table 2. Effect of increasing the number of sensors on the tsunami detection time in the case of the Cotabato Trench

method to a real-world problem of optimal sensors placement in the Cotabato Trench. We use the actual bathymetric profile of this trench to make the estimation of the wave travel time more accurate. One can observe that as the number of sensors increases, the detection time decreases. Moreover, the sensors become situated closer to the subduction zones. However, more sensors entail additional cost. Future works include setting cost and location constraints for the sensors.

We note that our method relies on an approximation of the wave velocity, but one can use the numerical solution of the 2D nonlinear shallow water equations for a more accurate computation of wave travel time.

5 Acknowledgment

The authors acknowledge the Department of Science and Technology – Science Education Institute (DOST-SEI) through the Accelerated Science and Technology Human Resources Development Program (ASTHRDP) Scholarship, for funding support.

References

1. Abe, I., Iwamura, F.: Problems and effects of a tsunami inundation forecast system during the 2011 tohoku earthquake. *Journal of Japan Society of Civil Engineers* **1**, 516–520 (2013)
2. Akbari-Jafarabadi, M., Tavakkoli-Moghaddam, R., Mahmoodjanloo, M., Rahimi, Y.: A tri-level r-interdiction median model for a facility location problem under imminent attack. *Computers and Industrial Engineering* **114**, 151–165 (2017)
3. Araki, E., Kawaguchi, K., Kaneko, S., Kaneda, Y.: Design of deep ocean submarine cable observation network for earthquakes and tsunamis. *OCEANS 2008 - MTS/IEEE Kobe Techno-Ocean* pp. 1–4 (2008)
4. Astrakova, A., Bannikov, D., S.G. Cherny, M.L.: The determination of optimal sensors' location using genetic algorithm. *Proc. 3rd Nordic EMW Summer School* **53**, 5–22 (2009)
5. Bautista, M., Bautista, B., Salcedo, J., Narag, I.: *Philippine Tsunamis and Seiches (1589 to 2012)*. Department of Science and Technology, Philippine Institute of Volcanology and Seismology (2012)
6. Braddock, R., Carmody, O.: Optimal location of deep-sea tsunami detectors. *Int. Trans. in Op. Res.* **8**, 249–258 (2001)
7. Dysthe, K., Bannikov, D., H. E. Krogstad, P.M.: Oceanic rogue waves. *Annual Review of Fluid Mechanics* **40**, 287–310 (2008)
8. Eble, M., Gonzalez, F.: Deep-ocean bottom pressure measurements in the northeast pacific. *Journal of Atmospheric and Oceanic Technology* **8**, 221–233 (1990)
9. Goda, K., Mori, N., Yasuda, T.: Rapid tsunami loss estimation using regional inundation hazard metrics derived from stochastic tsunami simulation. *International Journal of Disaster Risk Reduction* **40**, 101152 (2019)
10. Goda, K., Risi, R.D.: Probabilistic tsunami loss estimation methodology: Stochastic earthquake scenario approach. *Earthquake Spectra* **33**, 1301–1323 (2017)
11. Kaboli, S., Selvaraj, J., Rahim, N.: Rain-fall optimization algorithm: A population based algorithm for solving constrained optimization problems. *Journal of Computational Science* **19**, 31–42 (2017)

12. Khalid, A., Javaid, N., Mateen, A., Ilahi, M., Saba, T., Rehman, A.: Enhanced time-of-use electricity price rate using game theory. *Electronics* **8**, 48 (2019)
13. Lovholt, F., Kuhn, D., Bungum, H., Harbitz, C., Glimsdal, S.: Historical tsunamis and present tsunami hazard in eastern indonesia and the southern philippines. *J. Geophys. Res.* **117**, B09310 (2012)
14. Lovholt, F., Kuhn, D., Bungum, H., Harbitz, C., Glimsdal, S.: Historical tsunamis and present tsunami hazard in eastern indonesia and the southern philippines. *Journal of Geophysical Research* **117**, B09310 (2012)
15. Marchuk, A., Vasiliev, G.: The fast method for a tsunami amplitude estimation. *Math.Model. in Geoph.* **17**, 21–34 (2014)
16. Meza, J., Catalan, P., Tsushima, H.: A multiple-parameter methodology for placement of tsunami sensor networks. *Pure and Applied Geophysics* **77**, 1451–1470 (2020)
17. Mulia, I., Gusman, A., Satake, K.: Optimal design for placements of tsunami observing systems to accurately characterize the inducing earthquake. *Geophysical Research Letters* **44**, 12,106–12,115 (2017)
18. Nagai, T., Kato, T., Moritani, N., Izumi, H., Terada, Y., Mitsui, M.: Proposal of hybrid tsunami monitoring network system consisted of offshore, coastal and on-site wave sensors. *Coastal Engineering Journal* **49**, 63–76 (2007)
19. Omira, R., Baptista, M., Matias, L., Miranda, J., Catita, C., Careilho, F., Toto, E.: Design of a sea-level tsunami detection network for the gulf of cadiz. *Natural Hazards and Earth System Sciences* **9**, 1327–1338 (2009)
20. Park, H., Alam, M., Cox, D., Barbosa, A., van de Lindt, J.: Probabilistic seismic and tsunami damage analysis (pstda) of the cascadia subduction zone applied to seaside, oregon. *International Journal of Disaster Risk Reduction* **40**, 101152 (2019)
21. Pelinovsky, E.: Hydrodynamics of tsunami waves. Nihziini Novgorod, Institute of Applie Physics RAS p. 276 (1996)
22. Rabinovich, A., Eble, M.: Deep-ocean measurements of tsunami waves. *Pure and Applied Geophysics* **172**, 3281–3312 (2015)
23. Schindele, F., Loevenbruck, A., Herbert, H.: Strategy to design the sea-level monitoring networks for small tsunamigenic oceanic basins: The western mediterranean case. *Natural Hazards and Earth System Sciences* **8**, 1019–1027 (2008)
24. Valenzuela, V.P.B., Esteban, M., Takagi, H., Thao, N.D., Onuki, M.: Disaster awareness in three low risk coastal communities in puerto princesa city, palawan, philippines. *International Journal of Disaster Risk Reduction* **46**, 101508 (2020)