

# Probability-Based Rescue Model for Submersibles

## Summary

More than a century after the sinking of the Titanic, on June 19th 2023, a tourist submersible carrying five people disappeared near its wreckage, losing contact with the outside world. Despite extensive searches, no trace of it was found. In today's era of popular underwater exploration and tourism, it is of great significance to develop methods for quickly and accurately predicting the location of missing submersibles to facilitate timely rescue operations and prevent such tragedies from recurring. This paper attempts to find a way to minimize the search and rescue time.

We build **two main models: the location model and the searching model** to solve the problem. The location model predicts the position of the lost submersible and the searching model uses data provided by the location model to search for it. We also consider the equipment we need to prepare and how our model can be extrapolated to account for more complex situations.

Firstly, in the **location model**, we take many things into account, including the velocity of the current, the density of the water, the geography of the sea floor and the probability that the missing submersible suffers a propulsion loss. We also consider the survival instincts of people in the submersible. Uncertainty arises from the possibility of losing propulsion(**internal uncertainty**) and the complexity of the environment(**external uncertainty**). To lower external uncertainty, we advise the submersible to send its location and speed back to the host ship periodically. These numbers serve as a starting point in our model.

Then, in the **Prepare** stage of the problem, we evaluate different types of search and rescue equipment, dividing their costs into three parts: the cost of the equipment itself(availability), the cost of training personnel(usage) and the cost of storage and maintenance. Taking these costs into account, we decide to use **the portable sonar** in the searching process. For the process of salvaging submersibles, we equip the rescue vessel with mechanical arms and iron cages to secure the submersible awaiting rescue. Since the lost submersible is most likely deep in the ocean, we choose to use **submarine** equipped with the portable sonar to search for it underwater.

After that, in the **searching model**, The direction of the submarine movement **constantly changes** according to the current probability distribution calculated by the location model. We also take the accumulated search results into account by wiping the probability of the searched(and failed) regions to 0.

Finally, in the **Extrapolate** stage of the problem, we put forward our solutions to expand our model. To account for other locations, we plan to **update our geographic data**, including the velocity of the current, the density of water and the geography of the sea floor. As for when the number of submersibles increases, we accordingly increase the number of searching submarines and associate the submarines with lost submersibles, turning this new situation into our original one. We also put forward another idea in the *Future Work* section.

After constructing the model, we implement it with python code and achieve good results. Code tests prove the accuracy and efficiency of our model. Results can be seen in the *Model Implementation* section. In addition, our model is easy to implement and extend. By changing a few parameters in our code, we can stimulate more conditions of different sea areas, proving its good universality.

**Keywords:** Submersible Rescue; Probability Density; Convolution and FFT; Dynamic Prediction; Kinetics in Ocean

# Contents

<b>1</b>	<b>Introduction</b>	<b>3</b>
1.1	Problem Background . . . . .	3
1.2	Restatement of the Problem . . . . .	3
1.3	Our Work . . . . .	4
<b>2</b>	<b>Assumptions and Explanations</b>	<b>4</b>
2.1	Assumptions in terms of Geography . . . . .	5
2.2	Assumptions in terms of Kinetics . . . . .	5
2.3	Assumptions in terms of Probability . . . . .	6
<b>3</b>	<b>Notations</b>	<b>6</b>
<b>4</b>	<b>Model Preparation</b>	<b>7</b>
4.1	Data Collection . . . . .	7
4.2	Data Computation . . . . .	9
<b>5</b>	<b>Models</b>	<b>9</b>
5.1	Location Model . . . . .	9
5.1.1	Overall Introduction . . . . .	9
5.1.2	Detailed Model Design . . . . .	11
5.2	Preparation . . . . .	13
5.3	Searching Model . . . . .	14
5.3.1	Overall Introduction . . . . .	14
5.3.2	Detailed Model Design . . . . .	14
5.4	Extrapolating of the Model . . . . .	15
<b>6</b>	<b>Model Implementation</b>	<b>15</b>
6.1	Constants in Code . . . . .	15
6.2	Functions in Code . . . . .	16
6.3	Results . . . . .	17
<b>7</b>	<b>Strengths and Weaknesses</b>	<b>18</b>
7.1	Our Strengths . . . . .	18
7.2	Our Weaknesses . . . . .	18
<b>8</b>	<b>Future Work</b>	<b>19</b>
<b>Memo</b>		<b>20</b>
<b>References</b>		<b>22</b>

# 1 Introduction

## 1.1 Problem Background

People always have a great interest in sunken ships and jewels. To explore the mysteries of these wrecks, we need to build submersibles that can ensure the safety of personnel. Considering that submersibles may lose contact with the host ship at any time and sudden mechanical failures could lead to the loss of propulsion, we hope to design a method to better predict the location of the submersibles and initiate search and rescue operations as soon as possible.

However, changes in the submersible's location will continuously affect our predictions of its position, thereby altering the host ship's search trajectory for the submersible. We need to construct mathematical models to guide the host ship's course and the depth at which to deploy search for the submarine, in the context of constantly changing prediction results. Additionally, we must consider the cost of the search equipment and attempt to find a more economical solution.

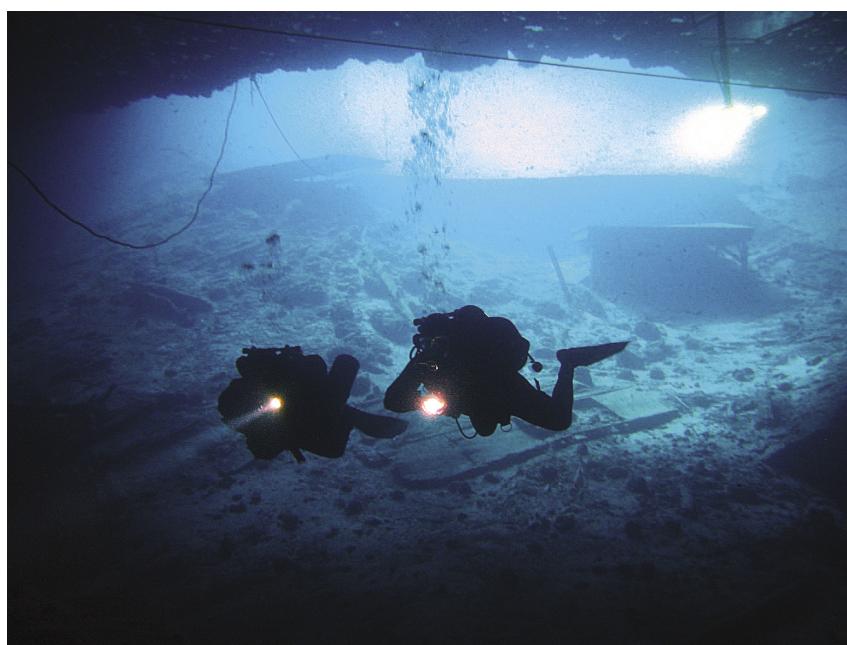


Figure 1: Underwater Exploration of Sunken Ships

## 1.2 Restatement of the Problem

Considering the background information and the specific constraints given, the re-statement of the problem can be expressed as follows:

- Develop models to predict the submarine's location over time, addressing uncertainties in predictions. Identifying data the submarine can send to the host ship to reduce these uncertainties and specify the necessary equipment for this data transmission.
- Recommend additional search gear for the host ship, considering the costs associated with availability, maintenance and usage.
- Create a model to suggest initial deployment points and search patterns, aiming

to minimize the time to locate a missing submersible. Adjust the likelihood of finding the submarine over time based on search efforts.

- Consider how to adapt the model for other destinations like the Caribbean Sea and for tracking multiple submarines in the same area.

### 1.3 Our Work

The problem requires us to find a better way to locate the submersibles as quickly as we can:

1. Based on the geographic data of the Ionian Sea and the information provided by the submersible before the loss of communication, a model to predict the **position of the submersible** is established;
2. The **probability distribution of the location** of the submersible in the near future is acquired, and special cases such as the submersible experiencing mechanical failure after losing connection to the host ship are considered;
3. Based on the assumptions of the velocity and location, this article effectively demonstrates the **accuracy of our model**, as well as the timeliness of the rescue work.

In order to avoid complicated description, and to intuitively reflect our work process, the structure of our work is shown in Figure 2:

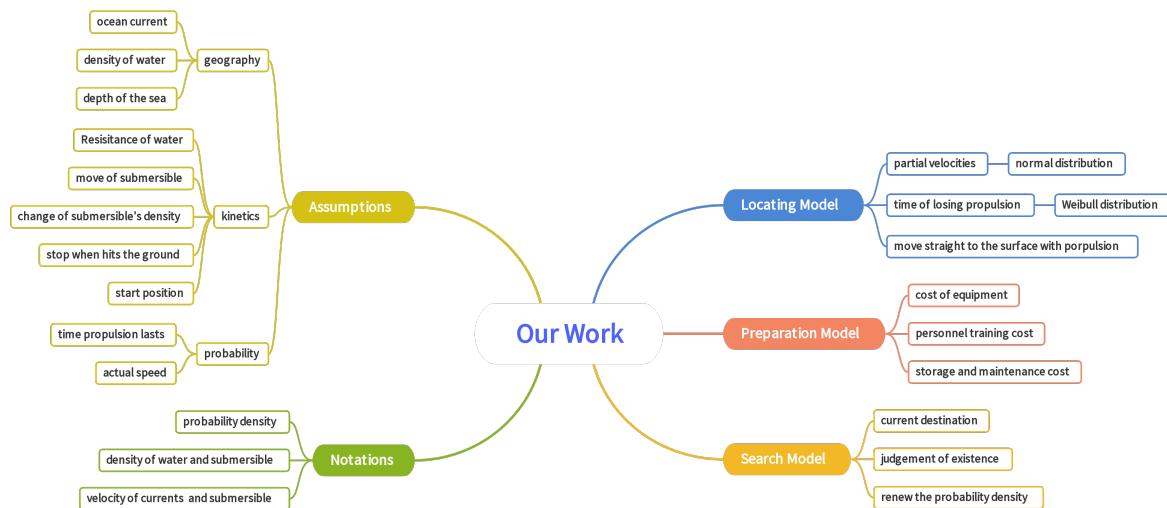


Figure 2: Structure of Our Work

## 2 Assumptions and Explanations

Considering the complex features of natural environment and the unpredictability of the submersible's precise speed, we have made a few reasonable assumptions to simplify the model beforehand. These assumptions fit into three categories, as shown below. Each hypothesis is closely followed by its corresponding explanation.

## 2.1 Assumptions in terms of Geography

**Assumption 1: The depth of the Ionian Sea at different horizontal positions is known.**

The bathymetric data of the Ionian Sea can be accessed online, so we assume the depths of the Ionian Sea at different horizontal positions are known values.

**Assumption 2: In the Ionian Sea, The density of water can be calculated.**

The density of the water is mainly determined by two factors: temperature and salinity. Both of which can be accessed online. We use TEOS-10[5] to calculate the density of water.

**Assumption 3: In the Ionian Sea, the velocity of the ocean currents can be calculated.[2, 3]**

In the framework of the EEA project BLUE-GREENWAY, ocean current conditions in the Ionian Sea have been studied for a period of ten years(2010-2019). There are already enough researches on the ocean currents, so we assume the velocity of the ocean currents can be calculated.

## 2.2 Assumptions in terms of Kinetics

**Assumption 1: Resistance is proportional to the relative velocity of the ship towards water.**

The submersible will most likely experience resistance in the water. In our model, the flow resistance follows the equation  $F_x = -k \cdot \Delta v_x$ . The same stands for y and z. Although this is not always the case, but compared to assuming resistance as a constant, what we use is usually a more accurate estimation.

**Assumption 2: From the moment that communication is lost, the submersible will move straight up with a given speed, until it is found or loses propulsion.**

The rescuers wish to minimize the time of the location of the submersible. **So do the people in the submersible itself.** Under the belief that the shallower the submersible is, the easier the location work will be, if there is still propulsion after the loss of communication, the submersible will move straight up.

**Assumption 3: From the moment that communication is lost, the submersible will change its density to a certain value.**

Under the same belief in Assumption 3, people in the submersible will change the density of the submersible to a certain value below the density of normal water in order to go up.

**Assumption 4: If at the end of a certain time period, the submersible touches the sea floor, its velocity will turn to zero.**

There may be scenarios in which the submersible hits the sea floor. In real life, this is probably a non-perfect elastic collision between the submersible and the Earth. However, as we do not know the collision factor, we simply assume the collision to be completely inelastic. This is equivalent to changing the velocity to zero whenever a collision occurs.

**Assumption 5: The searching submarine will be positioned at a fixed position at the beginning.**

Taking maintenance costs into account, the searching submarine cannot be at sea all the time, waiting for a rescue mission to appear. Thus, we assume that the searching submarine stays at shore most of the time. This includes the moment when the loss of communication occurs.

## 2.3 Assumptions in terms of Probability

**Assumption 1: The time that propulsion lasts(starting from the moment of communication lost) follows an Weibull distribution.**

As stated in the problem, the submersible may suffer a loss of propulsion. The Weibull distribution is used frequently in reliability engineering and failure analysis, according with our scenario nicely.

**Assumption 2: The actual speed of the submersible follows a normal distribution with a expectation of the given or calculated speed.**

*Explanation:* Due to the complexity of movement in water, after all our calculations, the actual velocity is ultimately uncertain. Even the velocity data sent back before the communication loss may be inaccurate. However, odds are that our calculations are not far from the precise numbers. That is why we assume the actual speed of the submersible to follow a normal distribution of the given or calculated speed.

These assumptions may be mentioned below again at the appropriate locations.

## 3 Notations

Some important mathematical notations used in this paper are listed in Table 1.

Table 1: Notations used in this paper

Symbol	Description
$x, y, z$	Coordinates of a point in a Cartesian coordinate system
$D(D \subset \mathbb{R}^3)$	The range in which we discuss the problem
$P(x, y, z, t)$	The predicted probability density of the lost submersible
$\rho(x, y, z)$	Density of sea water
$\rho$	Average density of the lost submersible
$\vec{v}_c(x, y, z)$	Velocity of currents
$\vec{v}_s$	Velocity of the searching submarine
$I_R(x, y, z)$	Equals to 1 if $(x, y, z)$ is in $R$ , 0 if not

\*There are some variables that are not listed here and will be discussed in detail in each section.

## 4 Model Preparation

In our model, we need some data such as the geography of the target sea area, the velocity of currents, etc.

### 4.1 Data Collection

The geography of the target sea area, that's to say the bathymetric data, and the temperature and salinity data can be accessed online. The temperature and salinity data are used to calculate the density of sea water.

Here are the database websites we can access data.

Table 2: Data and Database Websites

Database Names	Database Websites
Bathymetric Data	<a href="https://www.gebco.net/data_and_products/gridded_bathymetry_data/">https://www.gebco.net/data_and_products/gridded_bathymetry_data/</a>
Temperature and Salinity Data	<a href="https://www.ncei.noaa.gov/products/world-ocean-database">https://www.ncei.noaa.gov/products/world-ocean-database</a>

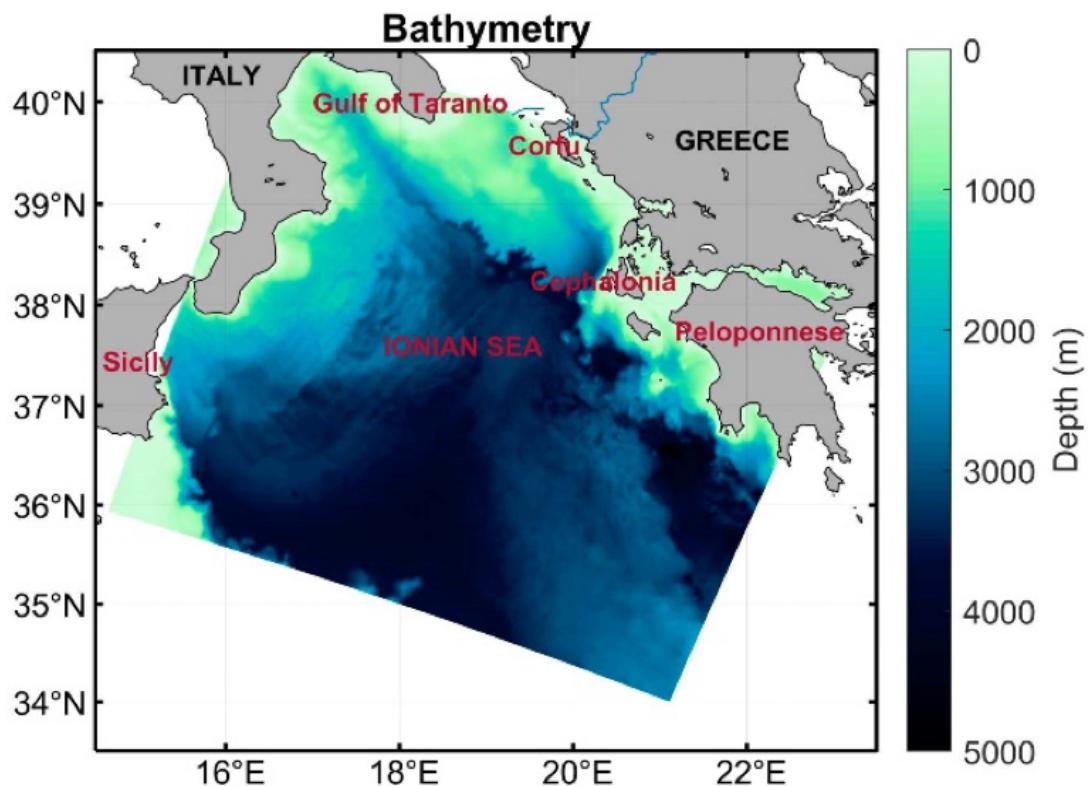


Figure 3: Bathymetry of the Ionian Sea[3]

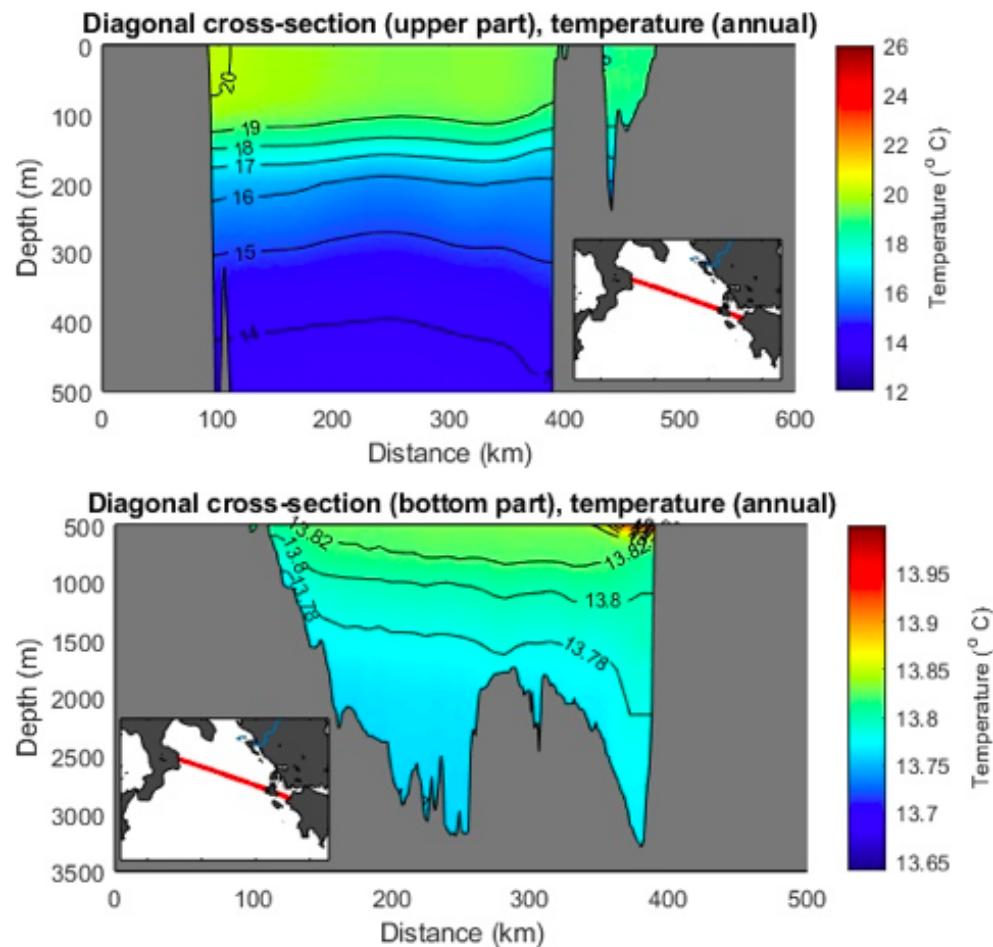


Figure 4: Mean annual values of temperature along depth[3]

## 4.2 Data Computation

The velocity of currents cannot be accessed directly online, and it changes over time. But we have methods to compute the velocity with data that can be accessed.

The velocity of currents can be estimated through the method given by current papers[2, 3], which we won't discuss in detail.

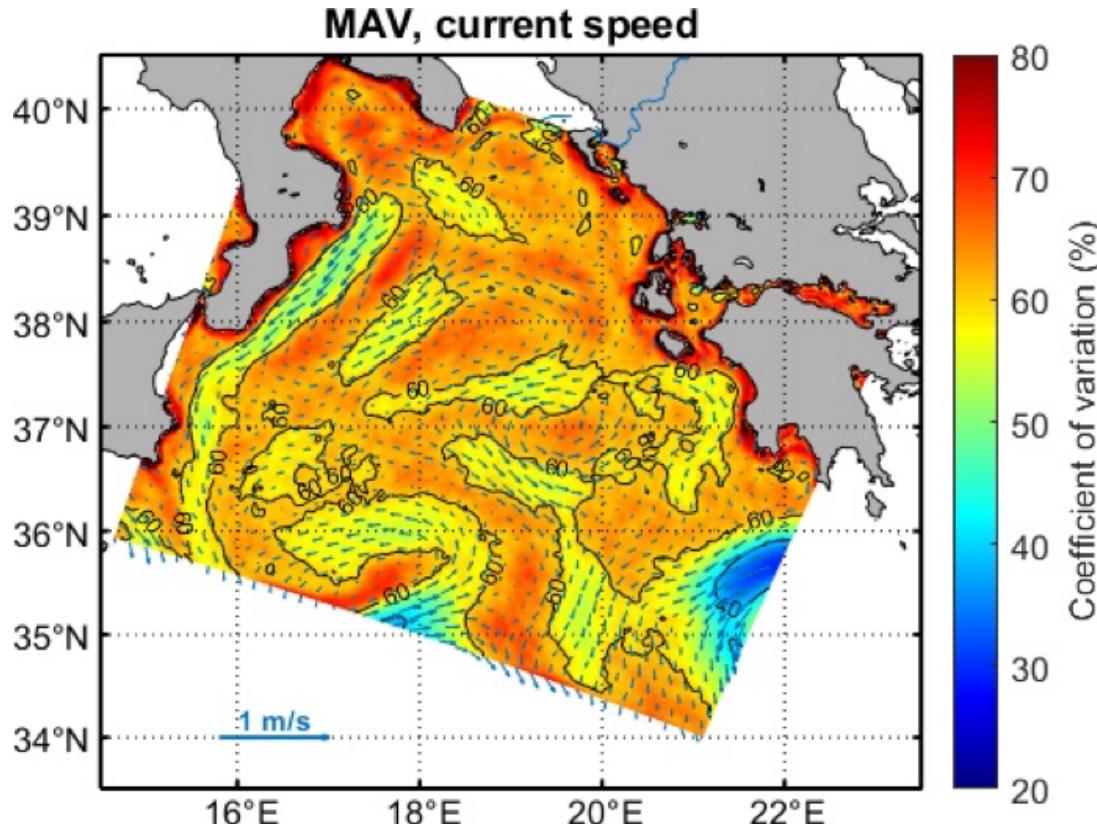


Figure 5: Mean annual variability of surface current speed[3]

## 5 Models

### 5.1 Location Model

In this section, we will discuss our model of predicting the position of the lost submersible.

#### 5.1.1 Overall Introduction

In the location model, we need to predict where the submersible is.

Firstly, we assume that the submersible periodically sends its position to the host ship before communication loss. So we know when it is the time it loses communication. To achieve this, we may need some equipment like sonar system, VLF radio communication system, etc.[6]

We will consider many external factors to make our model as accurate as possible, Here is the factors our model takes into account:

- Probability that the submersible loses propulsion
- Different water density at different positions
- Impact from currents
- Geography of sea floor
- We argue that predicting submersible velocities to obey a certain distribution to model the effects of other external uncertainties.

Considering the complex of the outer environment, we do not just predict an exact position. Instead, we use probability density to represent the prediction of its position, which changes over time.

See figure 6: A example of heatmap of probability distribution on a plane.

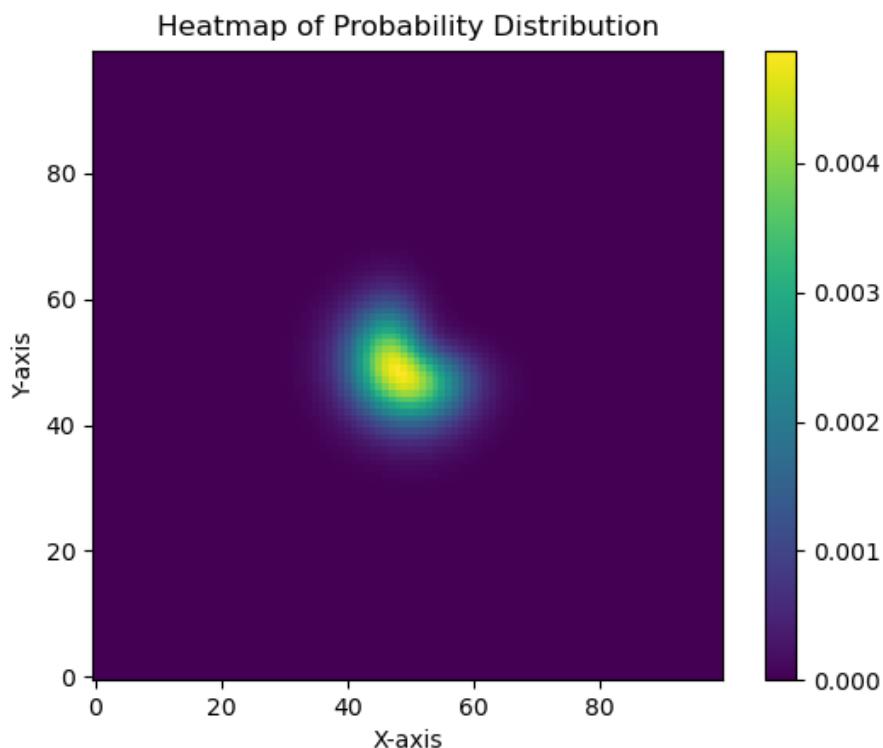


Figure 6: A probability distribution

To simulate the uncertainties of the external environment, we assume the predicted velocity is not a single value either. Instead, it obeys a normal distribution. And the average of the distribution is not a constant. It changes as the submersible is subjected to forces like gravity, buoyancy, as well as the thrust(or drag) by the currents.

Another important factor worth considering is the propulsion of the submersible. We can stipulate in advance that if its propulsion is not lost yet, then the submersible moves straight towards the surface, which is conducive to the rescue of the lost submersible as well as minimizing the cost of the rescue. Otherwise, when the propulsion is already lost, the submersible has to move by control of gravity, buoyancy and thrust(or drag) by the currents.

### 5.1.2 Detailed Model Design

Assume the probability of unit volume of the lost submersible being at  $(x, y, z)$  is denoted by

$$P(x, y, z, t)$$

which can be initialized by the last information received from the submersible before loss of communication.

Assume the predicted velocity of the submersible is

$$\vec{v} = [v_x, v_y, v_z]^T$$

Considering the complex external environment, we assume the partial velocities in all directions follow a normal distribution (denoted as  $\phi$ ), whose mean is respectively  $v_x, v_y, v_z$ .

Then we can find the probability density after a short time by the formula below:

$$P(x, y, z, t+\Delta t) = \iiint_D P(u, v, w) \cdot \phi(x-u-v_x \cdot \Delta t) \cdot \phi(y-v-v_y \cdot \Delta t) \cdot \phi(z-w-v_z \cdot \Delta t) \cdot dx dy dz$$

Then assume the probability that the submersible has lost propulsion is denoted by  $p(t)$ .

The Weibull distribution[4] is a commonly used probability distribution model for describing the reliability and lifetime distribution of systems. It particularly excels in characterizing the distribution of lifetimes for objects or systems. So we can assume the exact time when the submersible loses propulsion obeys a Weibull distribution.

$$t_0 \sim f(t; \lambda, k) = \begin{cases} \frac{k}{\lambda} \left( \frac{t}{\lambda} \right)^{k-1} e^{-(t/\lambda)^k}, & t \geq 0, \\ 0, & t < 0, \end{cases}$$

See figure 7 for image of Weibull distribution.

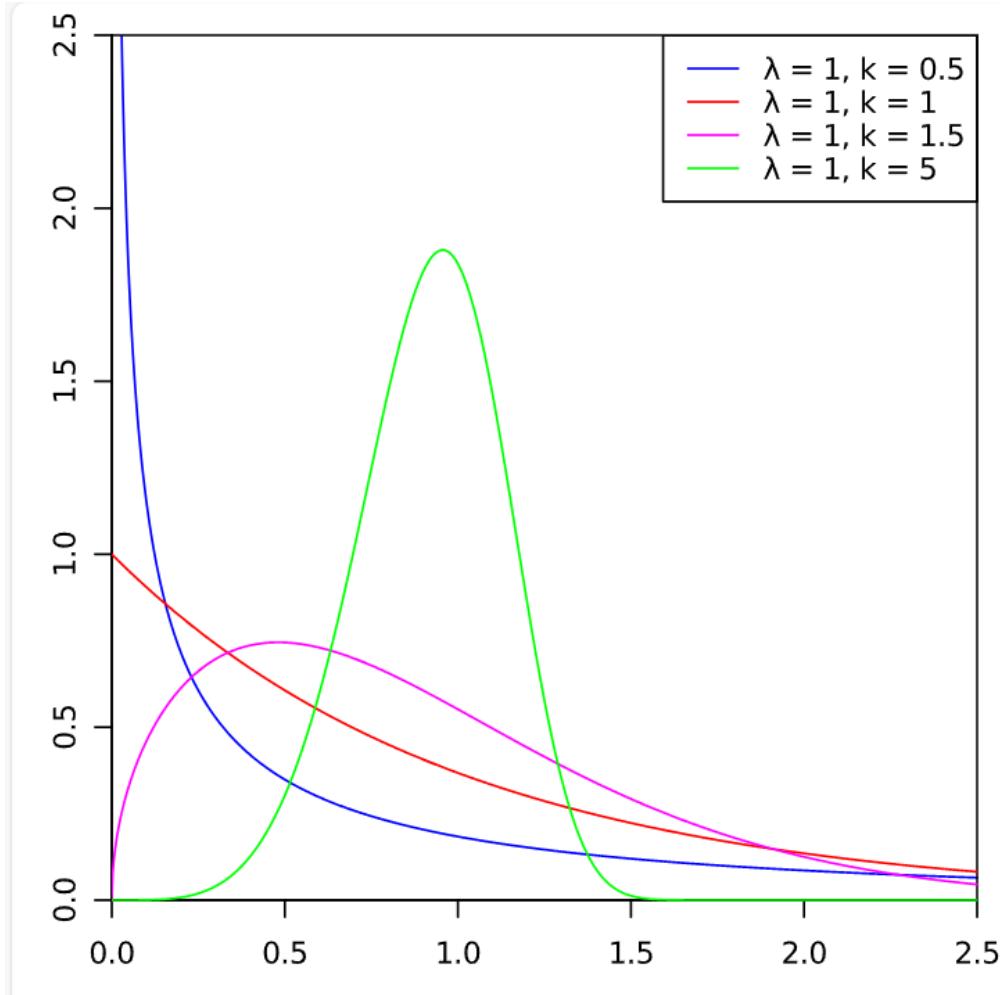


Figure 7: Weibull distribution

Then we can know

$$p(t) = \int_0^t f(s; \lambda, k) ds = 1 - e^{-(t/\lambda)^k}$$

To facilitate the rescue and save costs, we stipulate that if the propulsion is not lost, then the submersible moves straight towards the surface at a constant speed. When the propulsion is lost, the submersible moves with currents and gravity.

As we don't know whether the submersible has lost propulsion, we need to take both cases into account by their probabilities. Hence, we can calculate the velocity by the formula below:

$$\vec{v}(t + \Delta t) = \vec{v}_0[1 - p(t)] + [\vec{v}(t) + \vec{a}(t)\Delta t]p(t)$$

where  $\vec{v}_0$  stands for the constant upward speed when propulsion remains, and  $a(t)$  stands for the acceleration of the submersible when the propulsion is lost.

Now we focus on the acceleration of the submersible when the propulsion is lost. The forces on the submersible include buoyancy, gravity, and the thrust(or drag) of the current.

The gravity and buoyancy can be easily found as below

$$G = mg = \rho g V \quad F_b = \rho_0 g V$$

where  $\rho$  stands for the density of the submersible.

As every position has different density of sea water and we don't know the exact position of the submersible, so we use a specially chosen value  $\rho_0$  to represent the different value of water density. We choose  $\rho_0$  as the weighted average according to probability. That is

$$\rho_0(t) = \iiint_D \rho(x, y, z) P(x, y, z, t) dx dy dz$$

Here,  $\rho(x, y, z)$  stands for the density of the water at position  $(x, y, z)$ , and we use TEOS-10 to find  $\rho(x, y, z)$ . TEOS-10(Thermodynamic Equation of Seawater - 2010) is the international standard for the use and calculation of the thermodynamic properties of seawater.[5]

Then comes the thrust(or drag) of the currents, which is related to the velocity of the currents. We can assume the force by currents is proportional to the velocity difference, that is

$$\vec{F} = k(\vec{v}_s - \vec{v})$$

The way we choose  $\vec{v}_s$  is the same as how we choose  $\rho_0$ . We choose the special current velocity(to represent different values of currents' velocity)  $\vec{v}_s$  by

$$\vec{v}_s(t) = \iiint_D \vec{v}_c(x, y, z) P(x, y, z, t) dx dy dz$$

Here,  $\vec{v}_c(x, y, z)$  stands for the velocity of the currents at position  $(x, y, z)$ , and we can calculate  $\vec{v}_c(x, y, z)$  as mentioned in **Data Computation** section.

Then we have

$$m\vec{a} = \vec{F} + [0, 0, G - F_b]^T$$

From that we can find the accelerate and then find the velocity  $\vec{v}(t + \Delta t)$ .

Up to now we have considered different density of water, the probability of losing propulsion, and impacts of currents. Now we take the geography of the sea floor into account. When the submersible hits the sea floor, we can assume it loses velocity at once and it is not subject to horizontal force anymore as long as it doesn't leave sea floor.

## 5.2 Preparation

In this section, we will discuss our model on how to prepare, and the equipment to choose. To determine the equipment needed for search and rescue operations, we must consider the cost of various search and rescue devices. For any piece of equipment, the costs associated with completing a full search and rescue operation can be divided into three parts: equipment cost, personnel training cost, and storage and maintenance cost.

In the table listed below, we collect common search and rescue equipment available on the market and provided their prices(it is important to note that different models of search and rescue equipment and the region of purchase can affect their prices. For simplification, we use the prices in China and converted them into US dollars for calculation). All the data below are selected from China's major e-commerce platforms like alibaba and taobao, based on underwater sonar search requirements.

Equipment	Cost of Equipment	Personnel Training Cost	Storage and Maintenance Cost
Portable Sonar	\$2000	\$1000	\$500
Deep-sea Camera	\$8000	\$1500	\$800
Search and Rescue Drones	\$20000	\$2000	\$1000
Search and Rescue Underwater Robots	\$30000	\$2500	\$1200

Table 3: Cost of Common Equipment

Considering the total cost, we choose the portable sonar, which will be used when we need a submersible to seek for the lost one. As the sonar is much cheaper than other three equipment, a submersible equipped with a sonar can not only maximize the power of sonar to detect more things, but also save more money as both of them can be used for a long time with relatively low depreciation price.

For the process of salvaging submersibles, we generally use mechanical arms and iron cages to secure the submersible awaiting rescue. However, since the salvage operation is a necessary step by any method and the costs are essentially the same, a cost analysis is not required for simplification purposes.

### 5.3 Searching Model

In this section, we will discuss our model of how to search for the lost submersible.

#### 5.3.1 Overall Introduction

As the lost submersible may be deep in the sea, it's unreasonable to think that we can find the lost submersible from above the sea surface. We need something to dive into the sea and search for it. Thus, we assume we need a submarine to search into the sea, and the host ship to fetch the submersible once it is found.

We assume that the submarine can detect a range within which objects are located at a distance less than a certain value from it.

By the probability density of the submersible, we can decide where to go. And once we are certain the submersible is not in some range, then we let the probability of that range be zero. In this way, we could utilize the accumulated search results.

#### 5.3.2 Detailed Model Design

With the probability density of the submersible, we can choose a certain position  $(x_0, y_0, z_0)$  as the destination of the searching submarine. Here is how to choose

$$[x_0, y_0, z_0]^T = \iiint_D P(x, y, z, t) [x, y, z]^T \cdot dx dy dz$$

If the lost submersible is in the detecting range of the searching submarine, the submarine will inform the host ship to fetch the submersible. Otherwise we can update the probability density  $P$  by making the probability in the detected range zero. Here is how to do it:

$$P'(x, y, z) = \frac{P(x, y, z) \cdot (1 - I_R(x, y, z))}{\iiint_D P(u, v, w) \cdot (1 - I_R(u, v, w)) du dv dw}$$

Here,  $I_R(x, y, z)$  equals to 1 if  $(x, y, z)$  is in the detected range and equals to 0 if not.

## 5.4 Extrapolating of the Model

Our model can be extrapolated to different situations.

If the tourist destination is not Ionian Sea but Caribbean Sea(or any other areas), we only need to get the needed data of the area. The steps are the same as presented in **Model Preparation** section.

If there are more submersibles in the same general vicinity, we can find the probability density of each lost submersible. And we can also deploy more searching submarines into the sea. With the searching result of each searching submarine, we can update the probability density of each lost submersible in the same way presented in **Searching Model** section.

# 6 Model Implementation

In this section, we will briefly discuss how to implement our ideas with code.

## 6.1 Constants in Code

- What we use to calculate is the projections of the position, velocity and acceleration of the submersible on three axes. Therefore, we use 1d numpy arrays with three elements to characterize position, velocity and acceleration. We use the same method for the force.
- Although we possess the current velocity and water density at any given point, **the continuity of space** stands in our way, making it impossible to find an numpy array large enough. To make it possible to store geographic data in numpy arrays, we divide the sea into finite regions with volume, using a 1d numpy array *shape* with three elements to represent the distribution of the regions.
- We store the density of water in a 3d numpy array in the shape of *shape*. For the velocity of the ocean currents, we use a 4d numpy array. The first three dimensions represent the region, and the fourth dimension, consisting of three elements, represents the component of velocity on the x-axis, y-axis and z-axis. It is implied here that we assume the velocity of the ocean current and the density of water to be **constant in a region**.
- We use two boolean constants to represent whether the submersible still has propulsion, and whether the submersible is found.

- As for other constants, such as the mass of the submersible and gravitational acceleration, we simply use integers or decimals.

## 6.2 Functions in Code

- After a (failed) search, we need to update our probability distribution based on our search results. In brief, if we searched and failed, the regions we covered have a probability of zero. So what we need to do is multiply the probability of all the other regions by a decimal to ensure that the total probability is still 1. When we have changed the probability in the searched regions into 0, we use the function `np.sum()` to acquire the sum of the probability of all the remaining regions. `np.sum()` is the value that the probability of all the regions must be divided by in order to satisfy normalization. Rather than going through all the regions and updating them(if needed) one by one, we only check the regions within the searching range and turn a region's possibility into zero if it isn't zero already. This speeds up the process.
- As mentioned in the detailed design of the searching model, we choose a certain position according to the probability distribution at this time. The position is where the searching submarine will be headed towards. The idea of choosing the position is just like calculating the center of mass, only this time, we seek the center of probability. However, it will be too slow if we use three nested *for* loops. In our code, we use the \* multiplication of two numpy arrays, which is significantly faster. Take  $z_0$ , the vertical ordinate of the submersible's position for example. First we use a version of `np.sum()`, compressing the 3darray  $P$  into a 1darray that represents the probability for each value of  $z$ . Then we multiply this array by another 1darray. This array has the same number of elements, and the  $i$ -th element is  $i$ . Finally, we use `np.sum()` to calculate the sum of the result of the multiplication above, obtaining  $z_0$ . The process is similar for the calculation of  $x_0$  and  $y_0$ .
- To proceed from one probability distribution to the next, a **convolution** is needed. As we have turned the probability distribution into a discrete form by dividing sections, we use Fast Fourier Transforms to calculate the convolution. To obtain the probability density of the normal distribution, we import the function `norm()` from `scipy.stats`.
- To provide a **more intuitive understanding** of our model, we use the python package *Matplotlib* to generate figures of our model's output every second, starting from the moment of communication loss. Each figure is the combination of our location model and searching model, showing the current position(obtained with given or calculated speed) of the lost submersible, the current position of the rescue submarine and the probability distribution, all of them projected onto a two-dimensional plane. To accurately depict the positions and probability distribution, we project the results respectively onto the  $xy$  and  $xz$  planes each second.
- Other functions can be implemented in a relatively simple way, such as the function to determine whether we have found the submersible, and the function to update the velocity of the submersible with the submersible's acceleration.

### 6.3 Results

We use a set of constants to test our model. Below are our results:

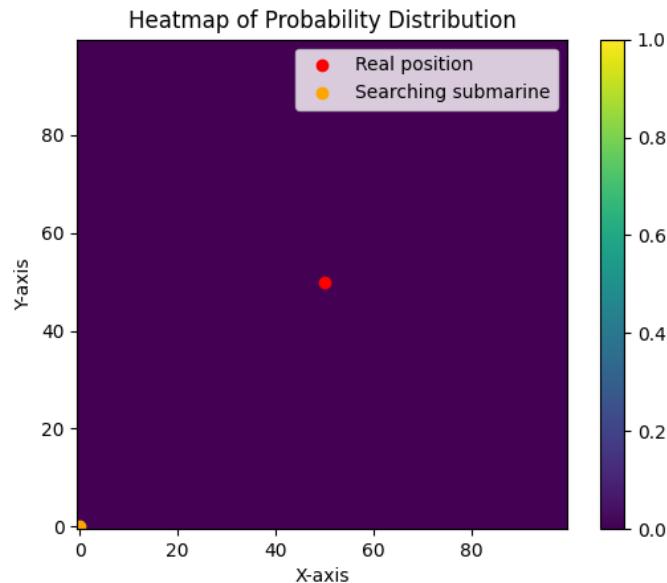


Figure 8: Generation at The Beginning

We can see in Figure 8 that at the beginning, the searching submarine is at the edge of the grid. Because the position of the submersible is determined at the beginning, so there is only one region with a probability of 1. For all the other regions, the probability is zero. The reason why the whole plane seems to be of probability zero is that the region where the probability is 1 is blocked by the red dot that represents the current position of the submersible.

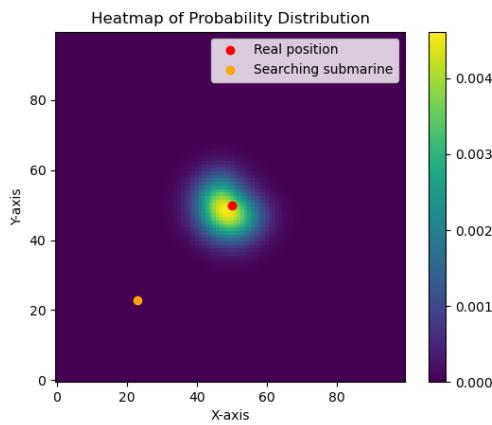


Figure 9: Generation after Some Time: xy

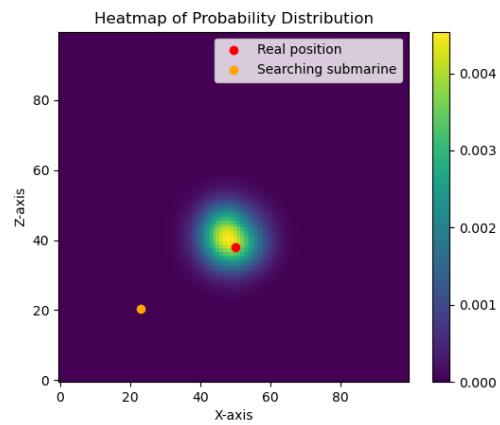


Figure 10: Generation after Some Time: xz

In Figures 9 and 10, we can see that after a few seconds, the probability distribution has spread out, but the real position is still not far from the regions where probability is highest. Meanwhile, the searching submarine has been moving steadily towards the submersible's real position. This can be seen on both the xy and xz planes.

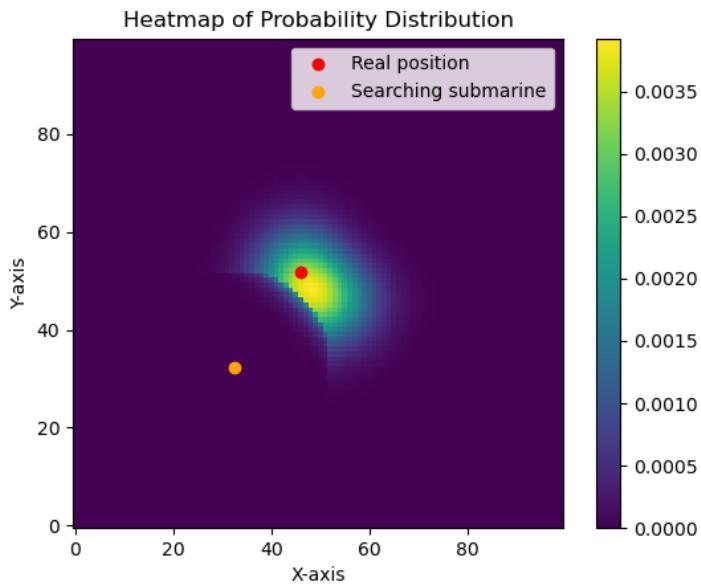


Figure 11: Generation When the Accumulated Search Results Intervened

The distribution shown in figure 11 looks sort of like an eclipse. This is because when the searching submarine did not find the submersible in regions with probability greater than zero, we updated the probability distribution based on this result. It's like we carve a sphere out of the probability distribution after every failed search. Note that the real position is still not far from the regions where probability is highest.

## 7 Strengths and Weaknesses

### 7.1 Our Strengths

1. **Rationality and Authenticity** Our model takes many factors into account. It's factual and reasonable. Our model considers the different density of water, impact of currents, geography of sea floor, probability of propulsion loss, and utilizes the accumulated searching results.
2. **Realizability** The data our model need can be either accessed online or accurately computed, as is demonstrated in the **Model Preparation** section. Our model can be implemented with code, as is shown in the **Model Implementation** section.
3. **Accuracy** Our model can accurately and quickly predict the position of the lost submersible. As we have taken many factors into account, our model is accurate enough to help rescue the lost submersible.
4. **Flexibility and Extensibility** Our model can be extended from Ionian Sea to other areas easily. All you need to do is to get the deserved data of the target area.

### 7.2 Our Weaknesses

1. **Complexity** Our model has many redundant computations. As is shown in the figure in **Model Implementation** section, there are a large number of regions whose probability densities are all zero, which will result in redundant computations.

2. **Overly Detailed** Our model is overly detailed in certain details that may not be very important. That will increase the computation burden.

## 8 Future Work

As mentioned in the section of our weaknesses, there is still room for improvement for our model. To begin with, although we realize the need for computation speed and set out to shorten the speed of our model, we still need much time for our calculations. In an emergency situation where every second counts, the model should be as fast as possible. Perhaps the use of certain data structures and algorithms can further shorten the computation time.

Next, **the initial position** of the searching submarine may deserve further consideration. In our model, we assume the initial position is  $(0,0,0)$ , but if we take the actual place of the sunken shipwreck to be visited into account, maybe we can find a place both at shore and at a closer distance for the searching submarine to wait for orders. However, due to the fact that we do not possess this data when solving the problem, we simply assumed the initial position to be  $(0,0,0)$ .

Finally, as stated in the **Extrapolate** section, there may be multiple submersibles in the same vicinity in the future. If more than one submersible needs to be rescued, our current approach is to send out multiple searching submarines, duplicating the location and search process of rescuing only one submersible. However, because the submersibles were originally in the same vicinity, this approach may result in the searching submarines clogging up, wasting detection space. Perhaps a more efficient approach will be to scatter the searching submarines a bit, at least making sure that their detection space does not coincide too much. This calls for the need of another model, but due to the limited time, we can only present our idea here.

## Memo

**To:** The Greek Government  
**From:** MCM Team #2419335  
**Date:** February 6th, 2024  
**Subject:** Probability-Based Rescue Model for Submersibles

---

Your Excellency:

In recent years, business activities about exploring are getting more and more popular. After the equipment of great detect devices, how to ensure the safety of the submersible becomes an urgent issue. Our team has developed a widely applicable model to achieve the purpose of minimizing the time to find the lost submersible. We appreciate the opportunity to introduce our strategy to you.

Considering the reaction of people staying in the submersible, the wisest choice for the captain is to go straight up, as the host ship know its current coordinate. Thus, the captain of the host ship can minimize the probable activity radius of the lost submersible and reduce the difficulty of finding it.

With this prerequisite, we can predict the most likely location of the lost submersible over time. When we provide data such as the depth, temperature, salinity, and the current speed of ocean currents at various points of the Ionian Sea, we can obtain from the model the most probable location of the submersible every second. Since the searching submarine scans the surrounding sea area continuously from the starting point, we can constantly update the heat map of the possible location of the submersible. The host ship always moves towards the average position weighed by probability on the current heat map. As our model employs a dynamic programming algorithm, we can continuously adjust the current optimal plan to ensure that the searching submarine does not blindly head towards locations with a lower probability, wasting precious rescue time.

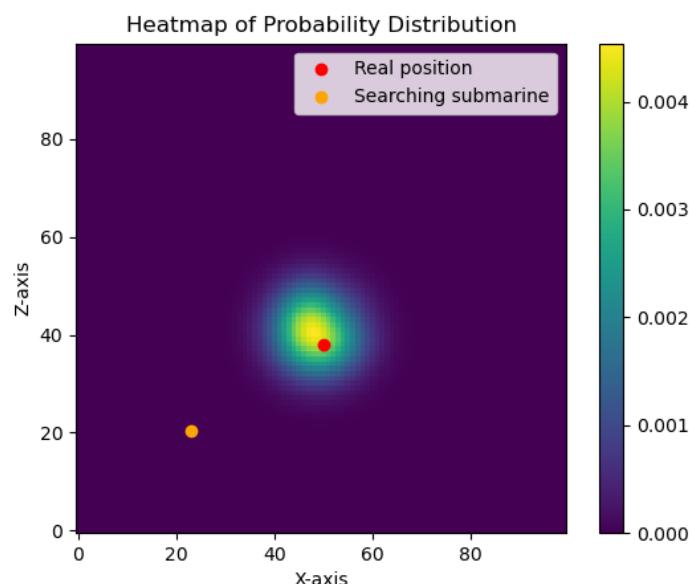


Figure 12: First Heat Map

In the first image, the real position of the missing submersible can be clearly seen near the center of the bright spot, indicating that our direction towards the center of the bright spot at this moment is accurate. Considering the radar's detection radius, we can almost ascertain the position of the missing submersible directly when the ship arrives near the center of the bright spot.

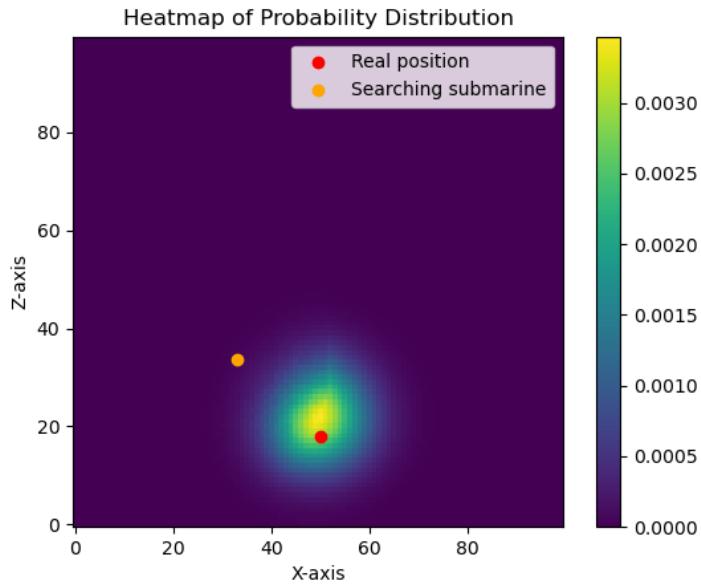


Figure 13: Second Heat Map

In the second image, the actual position is still locked near the center of the bright spot, which once again proves the rationality of our execution strategy. As seen, compared to the first image, the position of the bright spot has changed significantly. At this time, the dynamic programming algorithm allows us to adjust the route in time, turn around, and continue moving towards the red dot, shortening the search time.

Considering the urgency, timeliness of rescue, comprehensiveness of detection and cost of equipment, we recommend that ships conducting detection in the Ionian Sea use software with our model as the core to determine the position of the submersible based on real-time ocean data, search and rescue directions and other strategies. We are eager to help you ensure the safety of every submersible. If you have any further questions about our solution, please feel free to contact us. Our team members are willing to solve any questions you have and renew our model.

Hope we can keep cooperating.

Yours sincerely  
Team: #2419335

## References

- [1] Titanic tourist submersible: desperate search for sub missing with five onboard, The Guardian, <https://www.theguardian.com/uk-news/2023/jun/19/titanic-tourist-submarine-missing-north-atlantic>
- [2] Knutsen, Øyvind, Stefanakos, Christos, and Dag Slagstad. "Ocean current conditions in the Ionian Sea." Paper presented at the The 31st International Ocean and Polar Engineering Conference, Rhodes, Greece, June 2021.
- [3] BLUE-GREENWAY: Ocean current conditions in the Ionian Sea, Regional Funds Online Magazine, <https://regionalcoopmag.net/2021/08/22/blue-greenway-ocean-current-conditions-in-the-ionian-sea/>
- [4] Weibull distribution, Wikipedia, [https://en.wikipedia.org/wiki/Weibull\\_distribution](https://en.wikipedia.org/wiki/Weibull_distribution)
- [5] TEOS-10, Wikipedia, <https://en.wikipedia.org/wiki/TEOS-10>
- [6] How Do Submarines Communicate, Maritime page, <https://maritimepage.com/how-do-submarines-communicate/>