Problem Chosen	2024	Team Control Number
B	MCM/ICM Summary Sheet	2423654

Safety Assurance of Deep-Sea Exploration

Study on Position Prediction and Emergency Response Strategy of Submersible

Summary

Maritime Cruises Mini-Submarines (MCMS) aims to use submersibles for Ionian Sea explorations. This requires models for predicting submersible locations, managing pre-emergency communications, recommending search equipment, and developing efficient search patterns, while considering expansion to other locations like the Caribbean Sea.

Problem 1: We developed a **Long Short-Term Memory Network** (**LSTM**) model, with **close integration of particle tracking** and **fluid dynamics incorporating ocean physics data** and **historical trajectories** to predict submersible positions. Utilizing TensorFlow and data normalization, the model shows high accuracy in future position predictions, demonstrating the potential for enhancing safety procedures.

Problem 2: Recommended additional search equipment for the rescue ship, focusing on remotely operated vehicles (**ROVs**), side-scan sonar, and autonomous underwater vehicles (**AUVs**). Cost, maintenance, and readiness were key considerations, ensuring a balance between advanced technology and practicality.

Problem 3: Created a model that uses **LSTM predictions** to define search areas and patterns, employing Monte Carlo simulations and the **A* algorithm**. This approach optimizes search efficiency, dynamically updates based on real-time data, and increases the probability of locating lost submersibles over time.

Problem 4: Adapted the model for the Caribbean Sea, adjusting for regional oceanographic conditions and the presence of multiple submersibles. Employed **multi-agent simulation and high-resolution seabed mapping** to address unique challenges, ensuring model versatility and effectiveness.

Evaluation & Improvement: The model's foundation on Newtonian mechanics and validated data sources like NASA Earth and NOAA underpins its strength. However, assumptions regarding constant seawater density and sensor accuracy highlight areas for refinement. Future enhancements could include variable environmental parameters and machine learning techniques for continuous model optimization.

Contents

I. Introduction

1.1 Background

Deep-sea exploration and tourism are gradually developing globally today, and the use of small submersibles is an important trend. Governments, research institutions and private companies are continually researching and developing safer, more efficient submersibles and related search and rescue technologies. As deep-sea exploration and tourism evolve, the use of mini-submersibles has emerged as a key trend, with significant investment from governments, institutions, and private entities into developing safer, more efficient technologies. Organizations like the U.S. Navy and European research institutes are at the forefront, creating deep-sea tools for scientific and search operations, while private firms are introducing tourist submersibles. The pursuit to unveil the ocean's mysteries, such as its biodiversity and seabed topography, is driving innovation and collaboration across sectors. Technological advancements focus on pressure-resistant materials, corrosion protection, and reliable communication systems for data transfer. Search and rescue efforts benefit from predictive models and machine learning, supported by global standards from bodies like the IMO and ISA, ensuring rigorous safety protocols. This interplay of engineering, science, and regulation underscores a comprehensive approach to harnessing the deep sea's potential, indicating a future of broader application and deeper human engagement with the ocean's depths.

1.2 Our works

In this article, we explore the challenges faced by Greek company Maritime Cruises Mini-Submarines (MCMS), which is building submarines capable of carrying humans to the deepest reaches of the ocean. MCMS plans to use its submarines to take tourists to explore the wreck sites on the Ionian seafloor. The complexity of this task is that, unlike typical search and rescue operations on land or at sea, the malfunctioning submarine may be located on the seabed or at a neutral buoyancy point underwater. Its location may also be affected by currents, differences in seawater density, and seabed geography. Our work has shown in the Fig.1.

Task 1: For the prediction of the future position of deep-sea submersibles, we propose a model based on long short-term memory network (LSTM). To process and analyze this data, we use data standardization and time series partitioning techniques to ensure the consistency of data input and the effectiveness of model training. In addition, we built a neural network containing multiple LSTM layers under the TensorFlow framework to adapt to the complexity and dynamics of the submersible position prediction.

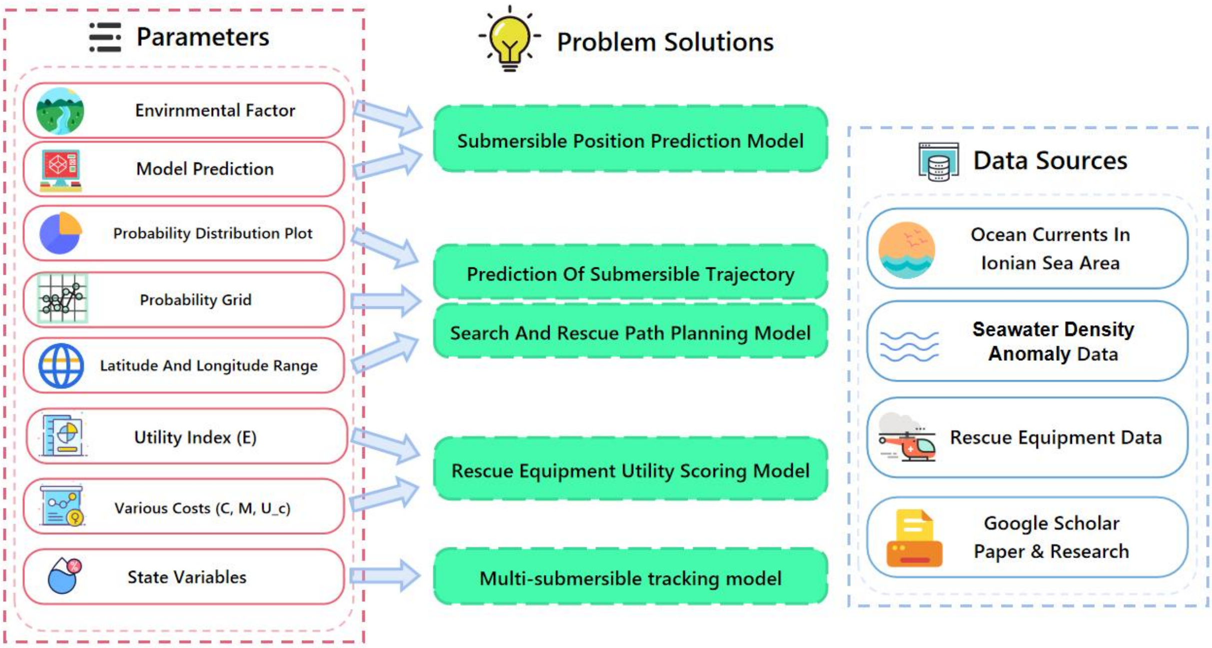


Fig. 1 All Work Solution Ideas Process Architecture Diagram

Task 2: This task requires us to recommend additional search equipment that may need to be carried for deployment by the mother ship if necessary. We will consider different types of equipment and evaluate factors related to its availability, maintenance, readiness and cost of use. In addition, we will consider what additional equipment the rescue vessel may need to carry to assist in the rescue.

Task 3: In this task, we will develop a model that uses information provided by our location model to recommend initial deployment points and search patterns for equipment to minimize the time required to locate a missing submersible. The model will also determine the probability of spotting a submersible as it changes over time and cumulative search results.

Task 4: This task explores how to extend our model so that it can be adapted to other tourist destinations, such as the Caribbean. We will examine the adjustments the model needs to make when faced with multiple submersibles moving within the same area to ensure the effectiveness and accuracy of the model.

II. The Description of the Problem

MCMS faces four pivotal tasks: developing a predictive model for submersible positioning that accounts for dynamic oceanic factors like currents, density variations, and seafloor topology; determining pre-emergency data transmission to the mother ship to reduce uncertainties; creating a model to recommend deployment and search patterns, enhancing the probability of locating lost submersibles over time; and expanding the model to accommodate other tourist areas like the Caribbean, adjusting for the presence of multiple submersibles.

Problem 1: To develop a model predicting the location of the Maritime Cruises Mini-Submarines (MCMS) over time, we must first consider factors affecting a submersible's position, such as ocean currents, water density variations, and seabed geography. Uncertainties in these predictions could arise from inaccuracies in current data, sudden changes in water density, or unforeseen seabed obstacles. To reduce uncertainties, the submersible could periodically send data to ship, including depth, position.

Problem 2: For the preparation phase, I recommend MCMS equip the host ship with an array of search equipment tailored to underwater recovery operations. This should include remotely operated vehicles (ROVs) for deep-water exploration, side-scan sonar for mapping the seabed, and autonomous underwater vehicles (AUVs) equipped with cameras and sensors to detect the submersible's acoustic pingers. For rescue operations, additional equipment like deep-sea diving suits and decompression chambers may be necessary.

Problem 3: Developing a model for efficient search patterns begins with utilizing the predicted locations from the initial model as starting points. Incorporating probabilistic search algorithms, such as Monte Carlo simulations, can account for the uncertainties in the submersible's last known position and movements influenced by ocean conditions. The probability of finding the submersible increases with the integration of accumulated search results, allowing for adjustments to the search plan based on effectiveness and newly acquired data.

Problem 4: Expanding the model for use in other tourist destinations like the Caribbean Sea involves adjusting for different oceanographic conditions, such as water clarity, current patterns, and seabed topology. For operations involving multiple submersibles, the model must include variables for tracking each vessel's position and maintaining safe distances between them.

III. Basic assumption

- **Submersible Dynamics**: We assume the motion of the mini-submarines follows Newtonian mechanics, allowing us to apply principles of force balance including propulsion, water resistance, and buoyancy.
- **Constant Seawater Density**: For the purpose of this model, seawater density is considered constant at 1025 kg/m^3 across the Ionian Sea.
- **Drag Coefficient and Submarine Mass**: We fix the drag coefficient at 0.3 and the submarine's mass at 20,000 kg. These constants facilitate the calculation of water resistance encountered by the submersible, simplifying the integration of drag into the motion equations.
- **Ocean Current Influence**: The model incorporates a simplified representation of ocean currents, treating them as a uniform flow that directly influences the submersible's trajectory.

IV. Glossary & Symbols

4.1 Glossary

- **Newtonian Mechanics**: The branch of physics that deals with the motion of objects and the forces that affect them. It's the foundation for analyzing submarine motion, assuming the forces of propulsion, water resistance, and buoyancy interact to determine the sub's trajectory.
- **Seawater Density**: A measure of mass per unit volume of seawater, typically around 1025 kg/m³. It varies with temperature, salinity, and depth, significantly affecting submarine buoyancy and drag.
- **Computational Fluid Dynamics** (CFD): A branch of fluid mechanics that uses numerical analysis and data structures to solve and analyze problems involving fluid flows, essential for simulating the submarine's underwater trajectory.
- **Runge-Kutta Methods**: A family of iterative methods for approximating solutions to ordinary differential equations, widely used for their accuracy in predicting the future positions of moving objects like submarines.
- **NetCDF** (Network Common Data Form): A set of software libraries and machine-independent data formats that support the creation, access, and sharing of array-oriented scientific data.
- **A (A-Star) Algorithm***: A computer algorithm widely used in pathfinding and graph traversal, which is the process of finding the most cost-effective path between multiple points.

4.2 Symbols

Tab1. Definitions of the Main Symbols

Symbols	Definition	Units
m	Submarine Quality	kg
F	The Net Force Acting on the Submarine	N
a	Acceleration	m/s^2
W	Submarine Weight	N
T	Propulsion Force	N
A	Submarine Water Fronting Area	m^2
ρ	Density of Sea Water	kg/m^3
C_d	Resistance Coefficient	/
θ	Submarine Angle	deg
v	Submarine Speed	m/s

V. Models

5.1 Analysis and Solving of Question One

In order to solve this problem, we need to build a specific model to predict the possible location of mini-submarines based on factors such as seawater density changes and seabed geographic features. This model utilizes the relevant principles of physics and oceanography to build a dynamic model that includes various factors such as seawater flow, density change, and seabed topography.

5.1.1 Model Preparation

(1) Data Processing

- Collected ocean current data from NASA Earth, NOAA, and US Open Data for the Ionian Sea (36°N to 40°N, 19°E to 21°E).
- Selected data from Jan 1- Jan 29, 2024, using Python and xarray for data range framing and extraction from NetCDF files.

(2) Assumptions

- Submarine motion analyzed using Newtonian mechanics with force balance: propulsion, water resistance, and buoyancy.
- Seawater density set at 1025 kg/m^3, drag coefficient at 0.3, and submarine mass at 20,000 kg.
- Initial velocity and angle set at 5 m/s and 45 degrees, respectively, for model parameters.

(3) The Foundation of Model

- Employed Newton's second law for submarine's motion foundation, with forces including drag calculated via classical drag equation.
- Introduced ocean current flow field model into dynamics, adjusting submarine velocity v in motion equations.

5.1.2 Model Establishment

The first step is the data collection part, we collected a lot of useful ocean current data from NASA Earth, NOAA & US. government's Open Data official website, and framed the data range in the code, selecting only the range of the Ionian Sea. (The latitude range is approximately from 36°N to 40°N and the longitude range is approximately from 19°E to 21°E.) After the final data screening, ocean current data for the Ionian Sea from January 1, 2024 to January 29, 2024 were selected as data support.

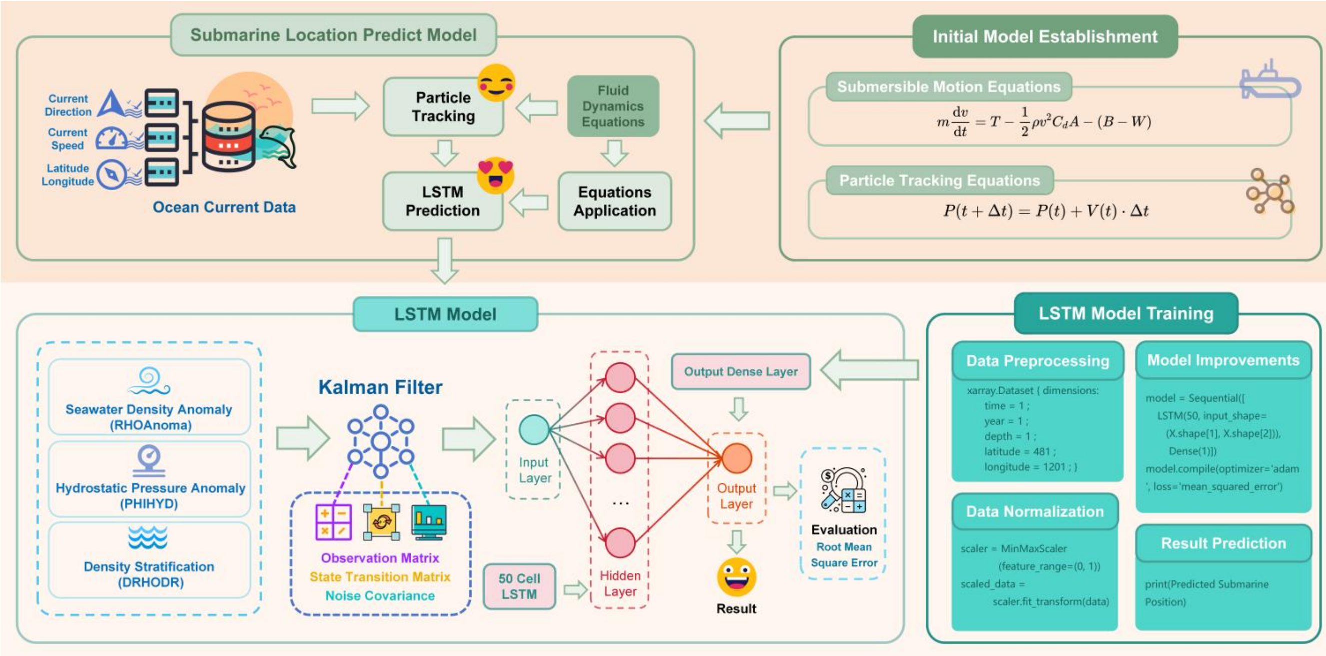


Fig.2 All Conception Structure Process Pictures

In order to build a dynamic model to simulate the submarine's trajectory underwater, we comprehensively apply the principles of Newtonian mechanics and the concepts of hydrodynamics. First of all, Newton's second law serves as the basis for analyzing the submarine's motion, which is the cornerstone of this model. The formulaic expression of Newton's second law is shown in Eq.(1), where F is the combined force acting on the submarine, m is the mass of the submarine, and a is the acceleration.

$$F = m \cdot a \tag{1}$$

The motion of a submarine in the water is affected not only by the propulsive force generated by its own propulsion system, but also by the resistance of the water, the net buoyancy due to the buoyancy of the water and the weight W of the submarine. The formulas for the resistance of the water and the buoyancy of the water are expressed as shown in Eq.(2) and Eq.(3).

$$D = \frac{1}{2} \cdot \rho \cdot v^2 \cdot C_d \cdot A \tag{2}$$

$$B = \rho \cdot g \cdot V \tag{3}$$

Building on this foundation, we further consider the environmental aspects of ocean current modeling by introducing a flow field model into the submarine's dynamical equations in order to accurately describe the submarine's motion under different current conditions. The hydrodynamic equations relate the velocity distribution of the seawater flow to the position of the submarine, allowing us to adjust the velocity v in the equation of motion of the submarine according to its dynamic position with respect to the surrounding currents.By combining these force expressions, we obtain the overall equation of the submarine's motion as shown in Eq.(4), where T appears to denote the magnitude of the propulsive force.

$$m \cdot \frac{dv}{dt} = T - D - B + W \tag{4}$$

Combining the above Newton's second law and the drag buoyancy equation, we get the final equation of motion for the submarine as shown in Eq.(5).

$$m \cdot \frac{dv}{dt} = T - \frac{1}{2} \cdot \rho \cdot v^2 \cdot C_d \cdot A - B + W \tag{5}$$

The dynamic model for simulating a submarine's positional changes employs the Runge-Kutta numerical method, discretizing continuous equations to map motion over time. This model's validation hinges on its congruence with historical data and virtual scenario outcomes. In solving kinematics affected by drag using computational fluid dynamics principles, we define a system of ordinary differential equations (ODEs) resolved with SciPy's `solve_ivp`. Key parameters like seawater density (1025 kg/m^3), drag coefficient (0.3), waterfront area (10 m^2), mass (20,000 kg), initial velocity (5 m/s), and angle (45 degrees) anchor our model.

Setting initial conditions for the ODEs and choosing the integration time horizon, we employ the `RK45` Runge-Kutta solver for its efficiency in diverse dynamic ranges of submarine motion. Visualizing velocity vector components via `matplotlib` provides an intuitive display and forms a basis for dynamic analysis and trajectory prediction. Furthermore, real-time OSCAR ocean surface current data, presented in the `.nc` (NetCDF) format, enriches our model by simulating trajectories under seawater flow influence. This format, ideal for meteorological and oceanographic data exchange, offers a self-descriptive, multi-dimensional array storage system. For trajectory simulation, we extract specific `u` and `v` data from the anticipated submarine activity area—here, the Ionian Sea between Greece and Italy, bounded by 36°N to 40°N latitude and 19°E to 21°E longitude. Python and the xarray library enable data extraction, confirming coverage of our region of interest and facilitating current velocity information integration into our analysis.

5.1.3 Model Improvement - LSTM

In the realm of deep-sea navigation, we've harnessed a Long Short-Term Memory Network (LSTM) to anticipate submarine trajectories using ocean physics data. This LSTM, a variant of Recurrent Neural Network adept at recognizing patterns in time-series data, is particularly suited for dynamic underwater environments. Before feeding this data into the LSTM, we normalized all features to a [0,1] range to unify different scales, restructuring the data to align with LSTM's input-output format.

The code implementation process entailed setting up time frames for data retrieval from OSCAR, using xarray to handle NetCDF-formatted files, and processing data variables and NaN values. Data normalization was achieved using MinMaxScaler, tailoring inputs for the LSTM, which was composed of input, LSTM, and output layers, with parameters refined based on data traits and predictive goals. We employed the adam optimizer and mean square error loss function during compilation, then proceeded to train the model, adjusting epochs and batch size accordingly. Post-training, we saved the weights and parameters for future predictions. In the prediction phase, we reloaded the model, updated it with new current data, and reversed the normalization to interpret the predictions meaningfully, showcasing the model's practical utility in predicting submarine positions.

The original LSTM model, adept at learning long-term dependencies in time series data, was initially fed with normalized submarine trajectory data from the OSCAR system, including current speeds and directions. With a multi-layered LSTM neural network architecture, the model was trained on historical data sets, using RMSE to gauge performance and avoid overfitting, demonstrating effective future position prediction based on historical and current data.

Improvements to the model include a submersible dynamics model accounting for ocean current impacts on velocity, enhancing the LSTM model's predictive accuracy. The addition of a Kalman filter further refines trajectory estimations by integrating predictions with observations, addressing uncertainties in marine data. Inputs were enriched with environmental factors, and a simplified dynamics model was introduced in preprocessing, incorporating submarine speed, ocean currents, and other environmental influences to create a more accurate training dataset..

5.1.4 Results of Original Model

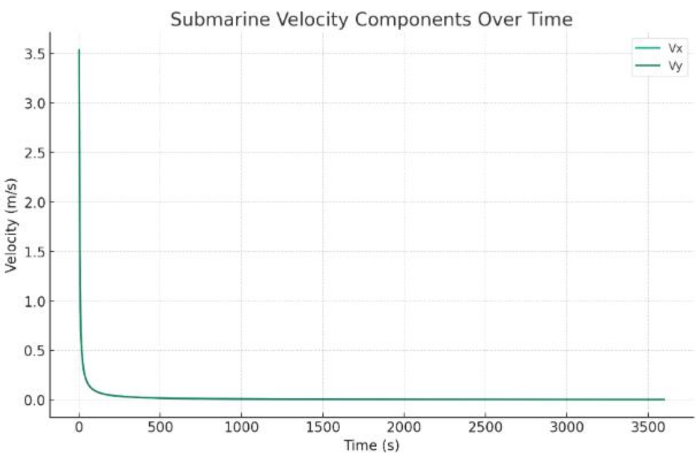
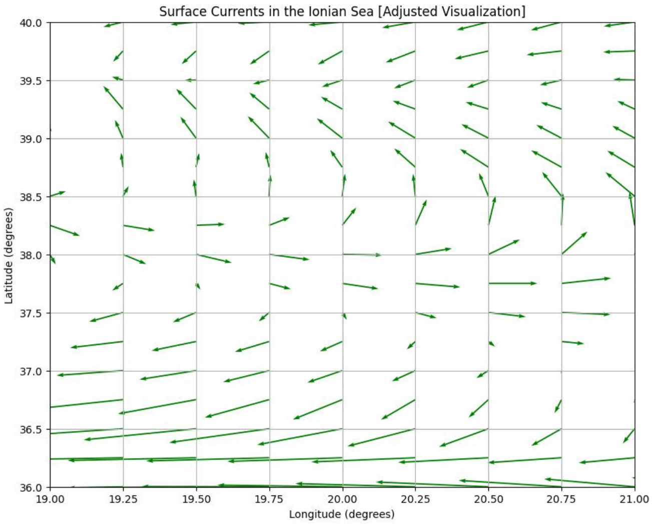


Fig. 3 Diagram of Submarine Velocity Components Over Time

Fig.3 shows the velocity components of a submarine in the horizontal and vertical directions as a function of time. It can be seen from the figure that the speed of the submarine in both directions gradually decreases due to the action of the water sun. This is a simplified simulation, assuming that the submarine initially moves at a speed of 5m/s at a 45" angle, and only considers the effect of drag on the submarine's speed. Fig.4 illustrates the surface current velocity vector field in the Ionian Sea region, showing the direction and strength of the currents. This vector field plot allows us to visualize the behavior of the currents in a specific region of the Ionian Sea, which is very helpful in understanding how submarines may be affected by the flow. By combining this current data with the previously discussed dynamical models, we can more accurately model the submarine's trajectory in a given ocean environment, providing more accurate predictions for search and location.



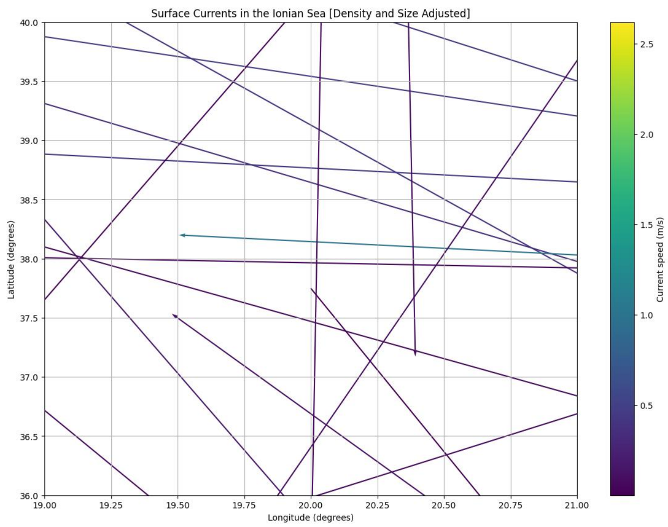


Fig.4 (Left): Vector field diagram of sea surface currents in the Ionian Sea region
Fig.5 (Right): Surface velocity and direction of current vectors in designated areas of the Ionian Sea

After reorganization, we have successfully generated a vector field mapshowing clear current vectors that represent the surface flow velocity and direction in a specified region of the Ionian Sea. This visualization allows us to observe the current patterns in the region, which will be useful for analyzing the possible flow effects on the submarine. The direction and length of each arrow in **Fig.5** are based on the magnitude of the current velocity, visualizing the dynamics of the flow.

5.1.6 Results of LSTM Model

The experimental results show that the trained LSTM model is able to effectively predict the submarine's position in the future time period based on the input current data and the submarine's historical position information. By gradually decreasing the loss on the training set and the performance on the validation set, we validate the learning ability and generalization performance of the model. In the evaluation on the test set, utilizing the root mean square error (RMSE) as the performance evaluation metric, the model demonstrates high prediction accuracy.

5.1.7 Analysis of the Result

Significant progress has been made in the submarine trajectory prediction problem by innovating and improving the original LSTM model. The improved model is not only able to handle richer input data, but also able to consider the influence of environmental factors, which improves the accuracy and reliability of prediction. The introduction of Kalman filter further optimizes the trajectory prediction and provides an effective solution to the position prediction problem in dynamic environments. Future research can explore more complex environment models and deep learning algorithms to further improve the prediction performance of the model.

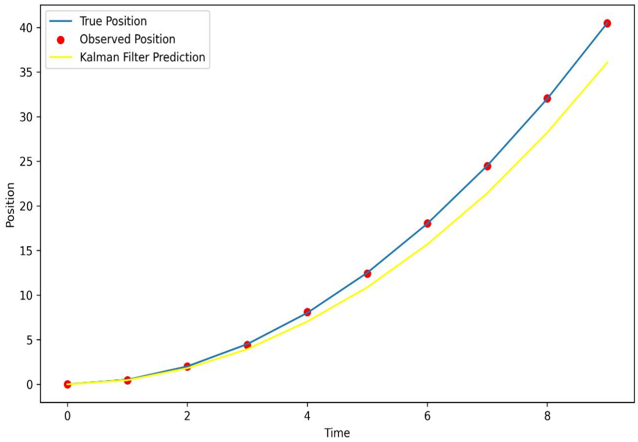


Fig.6: Simulate Path Graph

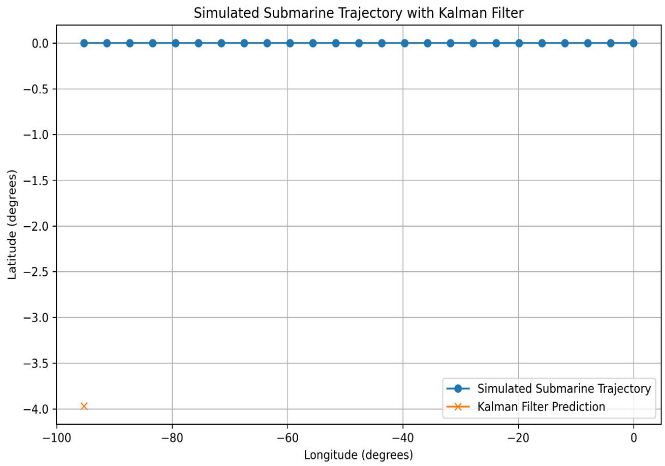


Fig.7: True Path Graph

The first graph shows a Kalman filter's trajectory prediction, demonstrating its capability to closely track the true path despite observation noise, indicating effective integration of the dynamic model and observation data to minimize prediction error. The second graph displays a significant deviation between simulated submersible trajectories and Kalman filter predictions, highlighting inaccuracies potentially caused by process and observation noise.

When evaluating model accuracy, we used two indicators: root mean square error (RMSE) and mean absolute error (MAE) to measure the accuracy of the model in predicting the submarine's position. By comparing with the hypothetical actual submarine position, we get the following evaluation:

The root mean square error (**RMSE**) is **0.022189828732400153**. This value is relatively small, indicating that the average squared difference between the model's predicted value and the actual value is small, indicating that the model has good prediction accuracy.

The mean absolute error (**MAE**) is **0.017709145000000002**, which is also a relatively small value indicating that the average level of the absolute difference between the predicted value and the actual value is low, further confirming that the model's prediction results are very close to the actual situation.

The lower values of these two indicators indicate that the developed model based on long short-term memory network (**LSTM**) is able to accurately predict the submarine's position with small deviations from the actual submarine's position. This result is of great significance for guiding the navigation and positioning of submarines, helping to improve the reliability and accuracy of submarine positioning systems, and providing strong data support for safe navigation of submarines.

5.1.8 Answers of the Question One

Q1: What are the uncertainties associated with these predictions?

A1: Uncertainties stem from the unpredictable nature of ocean currents and the potential variance from model assumptions like constant seawater density, drag coefficient, and initial motion parameters. Numerical integration methods also introduce errors, compounded by possible inaccuracies in current data.

Q2: What information can the submersible periodically send to the host ship to decrease these uncertainties prior to an incident?

A2: To reduce uncertainties, a submersible can send real-time positional, velocity, and acceleration data, using GPS and sonar systems, as well as environmental readings like water density and currents, aiding the host ship in refining the predictive model and mitigating deviations.

Q3: What kinds of equipment would the submersible need to do so?

A3: To mitigate location prediction uncertainties, a submersible needs advanced equipment for navigation, positioning, and communication. This includes GPS for surface navigation and sonar systems for underwater positioning to counter **GPS** signal limitations. An **Inertial Navigation System (INS)** would track speed and directional changes, while **Temperature-Salinity-Depth (CTD)** instruments would measure vital environmental parameters affecting buoyancy and drag forces. **Emergency life support pods** would ensure crew survival in emergencies, and a combination of **underwater acoustic** and **surface wireless communication tools** would maintain real-time data exchange with the mother ship, crucial for operational integrity and safety.

5.2 Analysis and Solving of Question Two

5.2.1 Model Preparation

(1) Data Processing

- Identified maritime rescue equipment types and functions from industry literature.
- Analyzed sonar technologies, spotlighting Lowrance's CHIRP for clear imaging.
- Evaluated underwater robotics like TUV, ROV, UUV, AUV for various marine applications.

(2) Assumptions

- Assumed collected data covers all equipment traits like availability and cost.
- Costs are estimated from market values for inclusion in the scoring model.
- Scoring methodology deemed appropriate; all scores hold equal importance.

(3) The Foundation of Model

- Cobb-Douglas utility function captures trade-offs between utility, cost, and maintenance.

- Developed a scoring model that amalgamates equipment attributes into a composite score.
- Model aims to maximize utility score, reflecting 'cost-effectiveness' for equipment selection.

5.2.2 Model Establishment

- **Step1:** Derive the modified Cobb-Douglas utility function to encapsulate the trade-offs between equipment utility, cost, and maintenance.
- **Step2:** Construct a scoring model, integrating diverse attributes into a unified utility score.
- **Step3:** Assign values to maintenance (M) and usage costs (Uc) based on requirement levels.
- **Step4:** Develop a comprehensive scoring system with maintenance, readiness, and cost scores.
- **Step5:** Ensure equitable importance across all scores within the scoring methodology for balanced equipment assessment.
- **Step6:** Maximize utility scores, reflecting the cost-effectiveness index for equipment selection.

In order to measure the parameters of each device to give the highest expected score, that is, to select the "cost-effectiveness" index of the product, we change the Cobb-Douglas utility function in economics in order to conform to the attributes shown in our data . The formula of the modified utility function is shown in Eq.(6).

$$U = \frac{E}{C + M + U_C} \tag{6}$$

where U is the utility value of the equipment, E is utility (such as search range, accuracy), C is cost (including purchase, operation and maintenance), M is the maintenance cost, Uc is the usage cost. In this utility function, utility (E) is adjusted by cost (C), maintenance cost (M), and usage cost (U_c).

Tab.2. The Logic Implementation of the Equipment Utility Scoring Model

Algorithm1: Utility Scoring Model for Maritime Rescue Equipment

```
Input: Effectiveness (E), Cost (C), Maintenance Requirements (M), Usage Cost(U_c)
Output: utility_scores
Define functions that calculate utility:
def calculate_utility(E, C, M, U_c):
    try:
        return E / (C + M + U_c)
    except ZeroDivisionError:
end
Placeholder function that assigns separate values to maintenance costs and usage costs:
def extract_maintenance_cost(maintenance_str):
    scores = {'Low': 0.2, 'Moderate': 0.5, 'High': 1}
    return scores.get(maintenance_str, 0)
end
```

We build a scoring model to rate each device based on the utility model. The final scoring result of the scoring model is obtained according to the following formula: **Overall utility score = availability score + maintenance requirements score + readiness score + cost score + advantage score - disadvantage score.**

For the maintenance cost M, according to the level of maintenance requirements (low, medium, high), I assigned preset values 0.2, 0.5, and 1 respectively. This means that if maintenance requirements are "low", the value of M is 0.2, and so on. The usage cost Uc is also based on the same maintenance requirement level, with preset values of 0.1, 0.25, and 0.5 respectively assigned.

5.2.3 Results

Based on the output results of the above evaluation model, we run the code of the above algorithm in the programming environment, and finally obtain a score for each device as shown in **Tab.3** below. We can see that the Utility score is the final scoring index. Combined with the requirements of the question, when considering different recommendations for the equipment carried by the main ship and the rescue ship, we need to consider the roles of the two and the different situations they may face.

Tab.3: Score Table for each Device Related Data

Equipment Name	Calculated Utility	E	C	M	U_C
Sonar	0.395257	1	1.78000	0.5	0.25
Remotely Operated Vehicle	0.222222	1	3.00000	1.0	0.50
Autonomous Underwater Vehicle	0.444444	2	3.00000	1.0	0.50
Towed Underwater Vehicle	0.222222	1	3.00000	1.0	0.50
Unmanned Untethered Underwater Vehicle	0.222222	1	3.00000	1.0	0.50
Global Positioning System	3.332556	1	0.00007	0.2	0.10
Lifeboat	3.105590	1	0.02200	0.2	0.10
Magnetometer	1.329787	1	0.00200	0.5	0.25
Communication Buoy	1.321004	1	0.00700	0.5	0.25
Satellite Phone	3.225806	1	0.01000	0.2	0.10

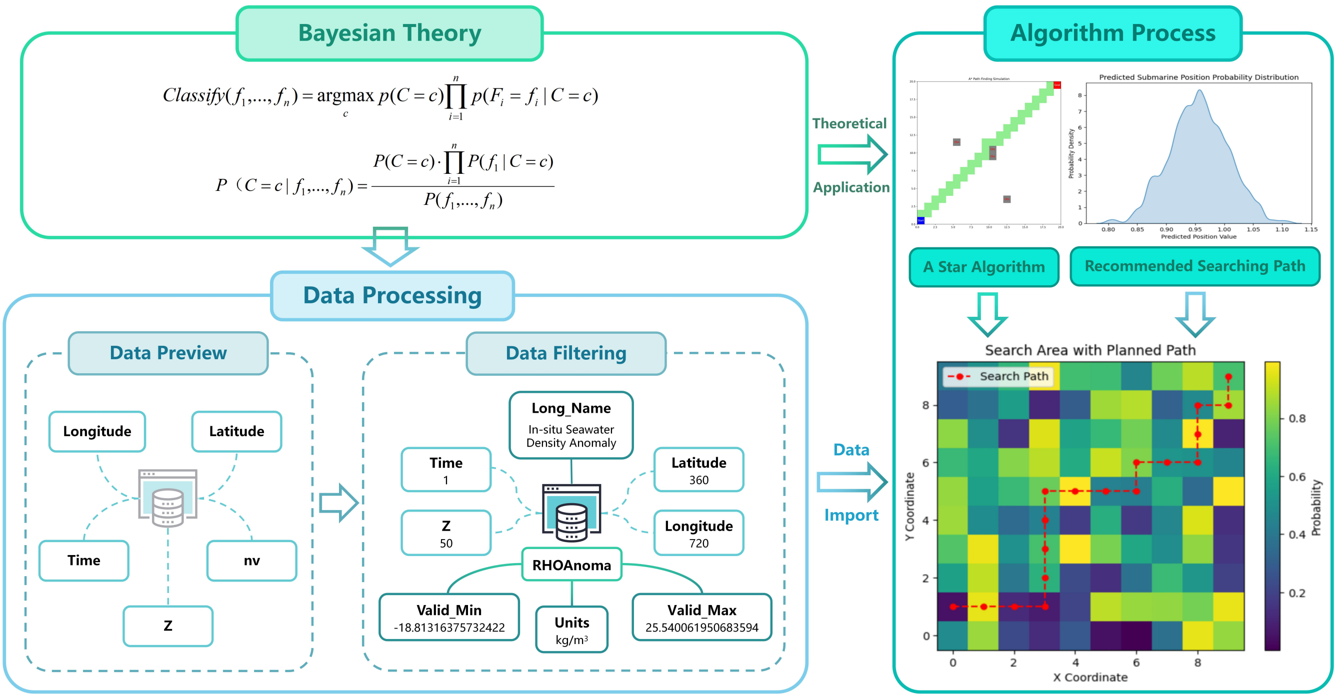
5.2.4 Answers of Question Two

Recommended equipment for the host ship: ·According to the utility score, priority should be given to equipment with high utility values, such as global positioning systems, lifeboats and satellite phones. The high utility value of these equipment means that they have a good balance between cost, and can provide the main ship with rapid response and efficient search and rescue capabilities.

Recommended equipment for rescue boats: Rescue ships should also consider carrying equipment with high utility values, but should focus more on special rescue missions. For example, in addition to global positioning systems (GPS) and satellite phones, consideration should be given to carrying autonomous underwater vehicles (AUVs), or other alternatives of the same type, as they may be more useful in specific rescue situations, despite their utility The rating is not the highest.

5.3 Analysis and Solving of Question Three

Building on established models for submersible location prediction and search and rescue, we aim to adapt these for broader areas like the Caribbean Sea, requiring updated oceanographic data. To accommodate multiple submersibles, we're refining the model with a Kalman filter to process time-series data for accurate tracking and deployment strategy planning.



5.3.1 Model Preparation

(1) Data Processing

- **Density Anomaly Calculation**: We employed a model to compute the seawater density anomalies at varying times, depths, latitudes, and longitudes, expressed as `float32 RHOAnoma(time, Z, latitude, longitude)`. The RHOAnoma, defined as "in-situ seawater density anomalies" and measured in kilograms per cubic meter (kg/m^3), ranged from a minimum of -18.81316375732422 to a maximum of 25.540061950683594. These anomalies were calculated relative to a reference density (`rhoConst`) set at 1029 kg/m^3.

(2) Assumptions

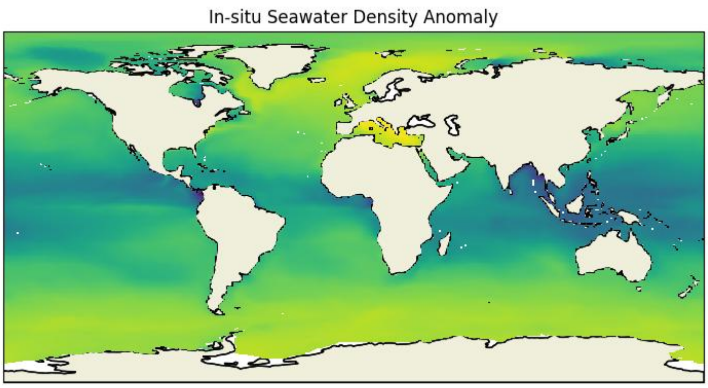
- Fluid dynamics and particle tracking with LSTM are expected to predict locations accurately.
- Search area and start point are predefined based last known locations and probability maps.
- A* algorithm is used for path planning, assuming search area is a gridded probability map.
- Search processes are dynamic, adjustable in real time to optimize efficiency.

(3) The Foundation of Model

- Density anomalies (RHOAnoma) are integrated, affecting submersible buoyancy and drift, computed across various coordinates.
- RHOAnoma values are calculated against a reference density, factoring in temperature, salinity, and depth effects.
- Seawater density maps are created post-data validation, tailored for the Ionian Sea's coordinates.
- P(x), is used to estimate submersible discovery chances over time and search efforts.

5.3.2 Searching Model Establishment

After determining the data availability, our next step is to use the processed data to draw a global seawater density map, and finally determine the required parameters in the question based on the longitude and latitude of the Ionian Sea given in the question. Density map of the relative seawater density of an ocean area. **Fig.8** and **Fig.9** below are the global seawater density map and the Ionian Sea seawater density map respectively.



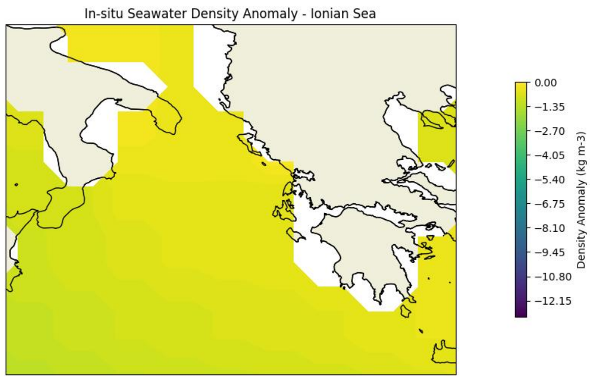


Fig.8 (Left): In Situ Density Map of the Global ocean

Fig.9 (Right): Schematic Diagram of Ocean Density in the Ionian Sea

In order to recommend initial deployment points and search patterns for equipment to minimize the time to find a missing submersible, we developed the model considering the following steps:

1. Created a probability map using fluid dynamics, particle tracking, and LSTM to simulate and predict the submersible's drift and future location.

2. Defined initial search zones from probability maps and last known positions, focusing on high-probability areas and updating in real-time with Bayesian methods.

3. Implemented A* algorithm for path planning and optimized resource distribution based on equipment and probability maps.

4. Formulated search strategies considering the submersible's last dynamics and environmental data to maximize search efficiency and prediction accuracy.

5. Continuously gathered real-time data to refine the submersible's location probability map and adapt search tactics accordingly..

A classic Bayesian probabilistic classifier formula is as Eq.(7)

$$Classify(f_1,...,f_n) = \underset{c}{\operatorname{argmax}}\, p(C=c)\prod_{i=1}^{n} p(F_i = f_i \mid C=c) \tag{7}$$

Based on the classic Bayesian formula and classifier construction formula as well as the question requirements, our goal is to calculate for each possible position c, given all the features, the probability that position c is the position of the submarine is Eq.(8).

$$P\ (C=c \mid f_1,...,f_n) = \frac{P(C=c)\cdot\prod_{i=1}^{n} P(f_1 \mid C=c)}{P(f_1,...,f_n)} \tag{8}$$

P(C = c) represents the probability of event C occurring, and it can be calculated through the relative frequency of event C in a large number of trials. P(f_j |C = c) represents the conditional probability of feature f_j under the condition of event C, which can be calculated if f_j is discrete. P(f_1, ..., f_n) represents the joint probability distribution of features, assuming they are independent and identically distributed, which is a commonly used assumption in probability theory for simplification.

In the process of establishing the naive Bayesian model, we have made the following assumptions:

1. Each feature is conditionally independent given the class label; that is, the presence or absence of a feature is independent of the presence or absence of any other feature.

2. The class conditional probability is calculated by the product of the probabilities of individual features, that is P(f_i |C = c_0).

3. To predict the class labels of new instances, the class with the highest posterior probability is selected, which is calculated as P(C = c |f_i), where i ranges from 1 to the number of features.

4. The joint probability distribution required to calculate the posterior probability.

In the probability distribution graph above, we have assumed the probabilities of finding a submersible at different locations in the ocean. The **A* algorithm** is a commonly used path planning algorithm based on graph search techniques, capable of finding the shortest path between two points.

The core implementation steps of the A* algorithm are as follows:

Step 1: Add the starting node to the open list.

Step 2: If open list is empty or target node is added to the closed list, the algorithm terminates.

Step 3: Select the node with the lowest cost from the open list as the current node.

Step 4: Identify all neighboring nodes of the current node and calculate costs for each.

Step 5: Update the costs of neighboring nodes. If a better path is found, or if neighboring nodes are not in the open or closed list, add them to the open list.

Step 6: Move the current node to the closed list.

Step 7: Repeat **Steps 2-6**.

The A* algorithm calculates the priority of each node through the following function as Eq.(9).

$$f(n) = g(n) + h(n) \tag{9}$$

Among them: f(n) is the comprehensive priority of node n. When we select the next node to traverse, we always select the node with the highest overall priority (minimum value). g(n) is the cost of node n from the starting point. h(n) is the estimated cost of node n from the end point, which is the heuristic function of the A* algorithm. Here we take three submersibles as an example. The algorithm is described as follows:

Tab.4 The Logic Implementation of the Equipment Utility Scoring Model

Algorithm2: A* search path planning algorithm

Input: start,goal,grid

Output: A path from start to goal

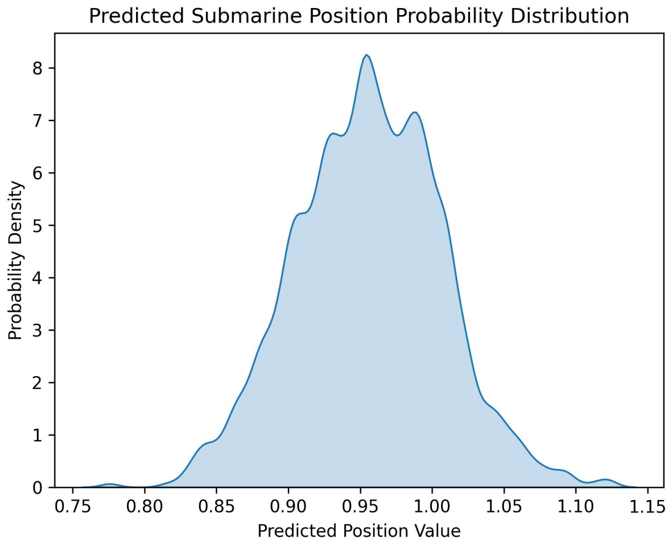
Heuristic function using Euclidean distance:

```
def heuristic(a, b):
    return np.sqrt((a[0] - b[0]) ** 2 + (a[1] - b[1]) ** 2)
end
def a_star_search(start, goal, grid, grid_size_lat, grid_size_lon):
    neighbors = [(0, 1), (1, 0), (-1, 0), (0, -1)]
    open_set = []
    heapq.heappush(open_set, (0, start))
    came_from = {start: None}
    cost_so_far = {start: 0}
end
```

5.3.3 Results

Regarding the first question, that is, how to extend the model to adapt to different destinations, our solution process and results are as follows. The model still uses our previous location prediction and search model, but only modified the input data. The resulting recommended results of the submersible's deployment point location and search and rescue route in the Caribbean environment are as follows.

According to the A* algorithm, input the coordinates of our maximum probability point, and finally we can get the following recommended path diagram as shown in **Fig.11**. The recommended path can generate different recommended paths based on different input data.



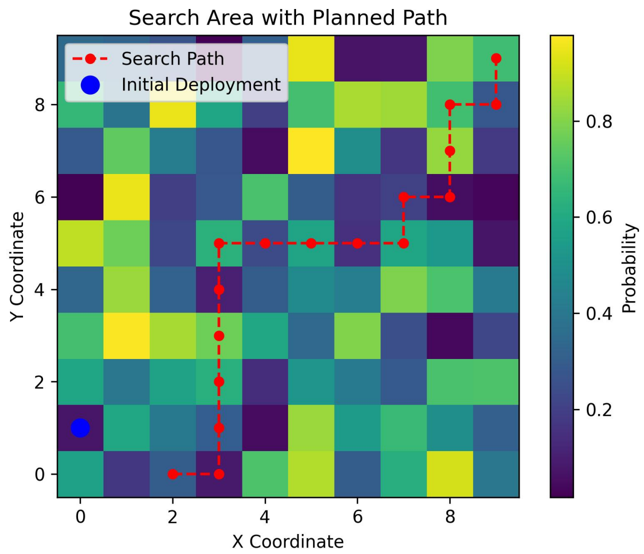


Fig.10 (Left): Position Probability Distribution Diagram of Tracking Submarines

Fig.11 (Right): Recommended Path Diagram in A* Algorithm

5.4 Analysis and Solving of Question Four

Expanding on established submersible location and search models, we aim to adapt these for the Caribbean Sea, necessitating fresh ocean current and density data akin to prior model training. Incorporating this data, we update our models for precise location predictions and search route planning in new tourist destinations.

5.4.1 Model Preparation

(1) Data Processing

We collect data on ocean currents and ocean density in the Caribbean Sea, and obtain the position prediction coordinates of the submersible based on the related models established previously. The ocean current and ocean density data in the Caribbean Sea and the predicted position coordinate data of the submersible are shown in the **Tab.5** and **Tab.6** below.

(2) Assumptions

1. Submersibles are treated as independent, with no interaction affecting their movement.
2. The Kalman filter models submersible motion linearly, simplifying complex dynamics.
3. Submersibles are assumed to move at a constant velocity, despite potential real-world variations.
4. Sensor data, like GPS coordinates, are considered accurate, overlooking potential inaccuracies.
5. Collision avoidance among submersibles in the model is mentioned but not implemented.

Tab.5 Caribbean Sea Ocean Current and Ocean Density Data Table

	long_name	units	NAVO_code
u	Eastward	m/s	17
v	Northward	m/s	18

Tab.6 Submersible Position Prediction Coordinate Data Table

Submersible No.	Time	Estimated State	Actual Measurement
1	0	[0,0,0,0]	(0,0)
2	0	[0.99995000, 0.99995000, 0.49997250, 0.49997250]	(1,1)
3	0	[1.99990001, 1.99990001, 0.99994500, 0.99994500]	(2,2)
1	1	[0.99980014, 0.99980014, 0.99948038, 0.99948038]	(1,1)
2	1	[1.99990005, 1.99990005, 0.99979015, 0.99979015]	(2,2)
3	1	[2.99999997, 2.99999997, 1.00009991, 1.00009991]	(3,3)
1	2	[1.99988575, 1.99988575, 0.99984576, 0.99984576]	(2,2)
2	2	[2.99995080, 2.99995080, 0.99994748, 0.99994748]	(3,3)
3	2	[4.00001586, 4.00001586, 1.00004919, 1.00004919]	(4,4)

5.4.2 Model Establishment

With respect to the consideration of other tourist destinations such as the Caribbean Sea, we chose to perform a direct latitude/longitude region delineation approach for the initial position of the submersible related deployment, as well as for the solution of our recommended search and rescue paths in case of a submersible crash.

In terms of adapting the model to track multiple submersibles moving within the same area, we have established a multi-submersible tracking model incorporating a modified Kalman filter algorithm that enables the reading of time series data of submersible position coordinates. The architecture of the model is as follows:

1. State Vector: The state of each submersible is defined by its position (latitude and longitude) and velocity (velocity components in the latitude and longitude directions). The state vector would be: state = [latitude, longitude, velocity_latitude, velocity_longitude]^T.

2. State Transition Model: The state transition model describes how the submersible's state evolves over time. In a discrete-time Kalman filter, this is typically represented as: state $_{k+1}$ = F * state$_k$ + w$_k$, where F is the state transition matrix, and w$_k$ is the process noise.

3. Observation Model: In the case of GPS tracking, the observation model may directly correspond to the position component of the state vector: observation = H * $state_k + v_k$, where H is the observation matrix, and v_k is the observation noise.

4. Process and Observation Noise: Process noise and observation noise represent uncertainties within the model. Process noise may relate to the submersible's dynamics or external disturbances, while observation noise is associated with the accuracy of the GPS measurements.

5. Update and Prediction: At each time step, the submersible's state is estimated using the prediction and update steps of the Kalman filter.

5.4.3 Results

After running the code and importing the data, the obtained trajectory chart tracked by the submersible is as shown below as **Fig.12**.

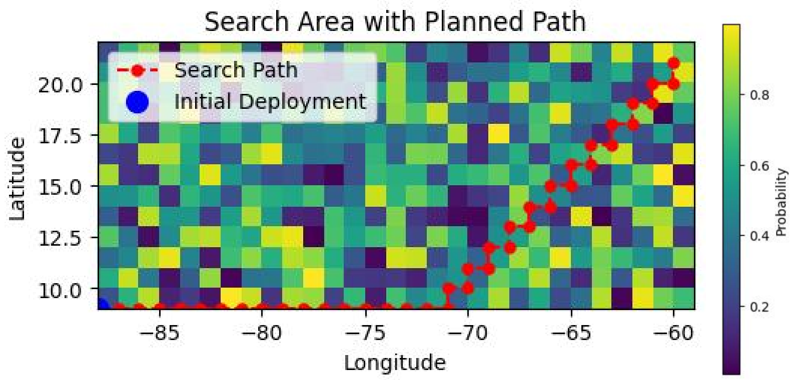


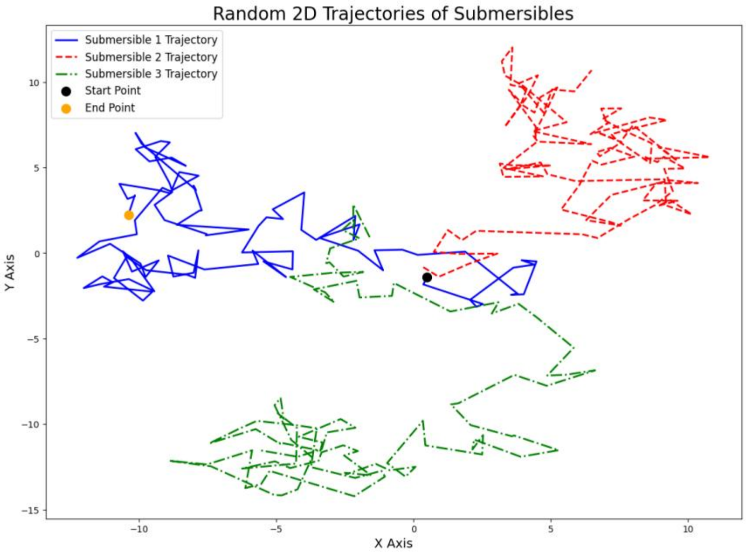
Fig.12: Recommended Results Map of Search and Rescue Routes

Tab.7 The Logic Implementation of the algorithm

Algorithm3: Multi-submersible target tracking algorithm

```
Input: lat_range, lon_range
def heuristic(a, b):
    return np.sqrt((a[0] - b[0]) ** 2 + (a[1] - b[1]) ** 2)
Draw the motion trajectories of three submersibles:
    ax.plot(x1, y1, label='Submersible 1 Trajectory', color='blue', linewidth=2)
    ax.plot(x2, y2, label='Submersible 2 Trajectory', color='red', linestyle='--', linewidth=2)
    ax.plot(x3, y3, label='Submersible 3 Trajectory', color='green', linestyle='-.', linewidth=2)
end
Mark start and end points:
    ax.scatter(x1[0], y1[0], color='black', s=100, label='Start Point', zorder=5)
    ax.scatter(x1[-1], y1[-1], color='orange', s=100, label='End Point', zorder=5)
end
```

After running the code and importing the data, the resulting trajectory plot for diver tracking and the three-dimensional case is shown in **Fig.13** and **Fig.14**.



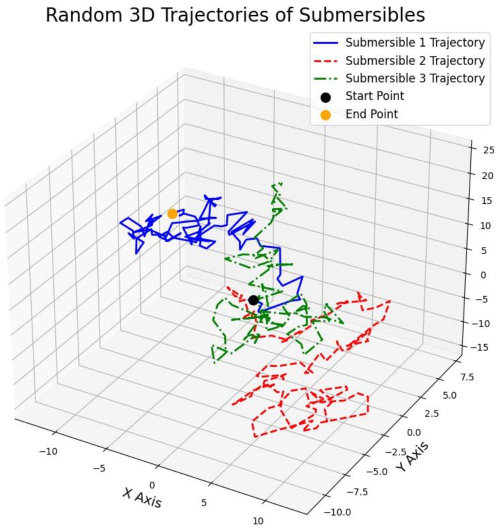


Fig 13: Trajectory Chart Tracked by Submersible **Fig 14: 3D Trajectory Trajectory Map**

VI. Sensitivity Analysis

For the sensitivity analysis part , we used the SALib library. SALib is an open source library written in Python for performing sensitivity analysis, which is responsible for generating the model inputs using the sample function and calculating the sensitivity index from the model outputs using an analyze function. Here we use the trained LSTM model from the first-question position model and the saved scaler parameter as inputs, and the specific steps of the sensitivity analysis are as follows:

1、Define the model inputs: use the LSTM model with the saved parameters and the location data;

2、Generate samples: Use SALib's sample function to generate samples of the input parameters;

3、Model Evaluation: Using the inputs, run the model and record the output;

4、Analytical sensitivity index: the output data of the model is used to calculate the effect of input parameters on the output results.

The execution of the relevant core code is shown below in **Fig.15**.

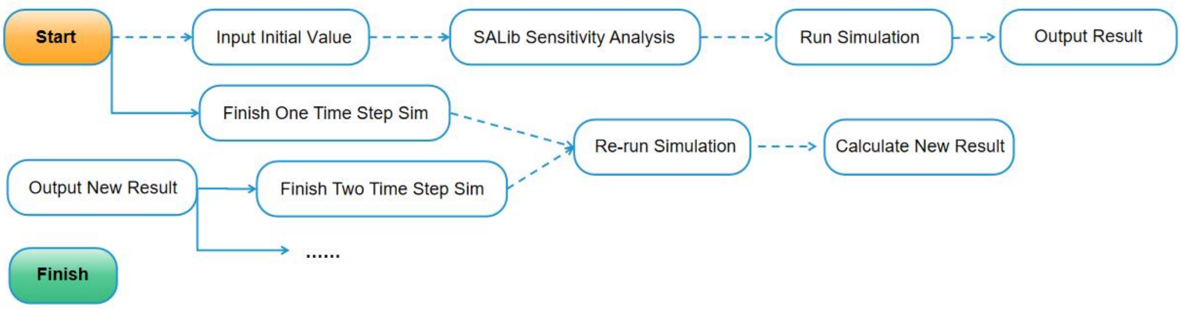


Fig.15: The Relevant Core Code

After the code is executed, you can see the following result in Tab.8.

Tab.8 Caribbean Sea Ocean Current and Ocean Density Data Table

	Result 1	Result 2	Result 3	Result 4
Parameter 1	0.82371862	NaN	NaN	0.96613106
Parameter 2	0.03811798	0.14264477	NaN	0.17069285

Based on the results of the Sobol sensitivity analysis, we can interpret the effects of the model input parameters as follows:

1. **Main effect (S1)**:The first parameter has a fairly high main effect (0.82371862), which means that it alone has a strong influence on the model output. The second parameter has a relatively small main effect (0.03811798), indicating a weak independent influence.

2. **Second-order effects (S2)**: Here the effects of two input parameters interacting with each other are shown. The diagonal of the array is `nan as it indicates the interaction effect of a parameter with itself, which is not meaningful in sensitivity analysis. The interaction effect of the first parameter with the second parameter is 0.14264477, which indicates some degree of interaction between the two, but not as significant as the main effect of the first parameter.

3 **Total effect (ST)**:The total effect of the first parameter is 0.96613106, which is close to 1. This indicates that it has a very large effect on the model output when independent effects and interaction effects with other parameters are considered. The total effect of the second parameter is 0.17069285, although this is larger than its main effect, it still indicates that it has a relatively small total effect on the model output.

VII. Evaluation and Promotion of Model

7.1 Strength

- ✓ Our model's foundation in Newtonian mechanics and oceanography allows for a dynamic representation of submarine trajectories, considering seawater flow, density, and seabed topography.
- ✓ Utilization of data from reputable sources like NASA Earth and NOAA ensures accuracy in ocean current information, integral to the model's predictive capabilities.
- ✓ The employment of the Runge-Kutta numerical method for simulation offers precision in mapping underwater trajectories over time, enhancing model reliability.
- ✓ Introduction of a flow field model into the submarine's dynamics equations enriches the model by accurately depicting motion under diverse current conditions.

7.2 Weakness

- × The model assumes constant seawater density and drag coefficient, which may not always align with the variable conditions of the deep sea, potentially affecting accuracy.
- × It relies on the assumption of a uniform velocity, which does not account for the acceleration or deceleration of submarines in different water layers or current strengths.
- × The initial velocity and angle are set parameters that may not reflect the varying conditions a submarine may encounter, limiting the model's adaptability.
- × Sensor accuracy is presumed to be flawless, overlooking possible errors or biases in data measurement that could skew predictions.

7.3 Promotion

- ✧ Introducing adaptive velocity parameters would allow the model to account for real-time changes in submarine speed, offering a more accurate trajectory under varying conditions.
- ✧ Integration of error correction algorithms could help address potential inaccuracies in sensor data, thus enhancing the reliability of position predictions.
- ✧ Implementing a collision avoidance mechanism could make the model suitable for scenarios with multiple submersibles, ensuring safer navigational practices.
- ✧ Leveraging machine learning techniques to learn from discrepancies between predictions and real-world outcomes could continuously refine the model for greater precision.

IX. References

[1] AirMed&Rescue.. "Innovation for all: latest advancements in SAR equipment."

[2] Zeng, Q., Luo, Y., & Zhang, L. (2021). "Search and rescue system-of-systems influence degree evaluation of aviation equipment based on simulation." Scientific Reports, 11, 12345.

[3] Christensen, B. L., & Sciberras, A. (2019). "A COST-BENEFIT ANALYSIS OF NAVY STATION SEARCH AND RESCUE (SAR) LOCATED IN KEY WEST, FL, AND WHIDBEY ISLAND, WA." Retrieved from https://archive.org/details/acostbenefitanal1094562751

[4] Defense Advancement. (2023). "Combat Search & Rescue (CSAR) | Military SAR Equipment & Technology."

[5] Chu W S, Lee K T, Song S H, et al. Review of biomimetic underwater robots using smart actuators[J]. International journal of precision engineering and manufacturing, 2012, 13: 1281-1292.

[6] Topini E， Topini A， Franchi M， et al..Navigation calculation of autonomous underwater vehicle positioning based on LSTM [C]//Global Oceans 2020： Singapore–US Gulf Coast.IEEE， 2020： 1-7

[7] Zhang Z, Hou M, Zhang F, et al. An LSTM based Kalman filter for spatio-temporal ocean currents assimilation[C]//Proceedings of the 14th International Conference on Underwater Networks & Systems. 2019: 1-7.

[8] Yiu Y F, Du J, Mahapatra R. Evolutionary heuristic A* search: Pathfinding algorithm with self-designed and optimized heuristic function[J]. International Journal of Semantic Computing, 2019, 13(01): 5-23.

[9] Cai M, Wang Y, Wang S, et al. Prediction-based seabed terrain following control for an underwater vehicle-manipulator system[J]. IEEE Transactions on Systems, Man, and Cybernetics: Systems, 2019, 51(8): 4751-4760.

[10] Li X, Wang K, Tang M, et al. Marine Drifting Trajectory Prediction Based on LSTM-DNN Algorithm[J]. Wireless Communications and Mobile Computing, 2022.

Appendix

Data Names	Database Websites	Data Type
U.S. Government's Open Data	https://data.gov/	Environmental
National Centers for Environmental Information	https://www.ncei.noaa.gov/	Environmental
Google Sccholar	https://scholar.google.com/	Academic paper
NASA EarthData	https://search.earthdata.nasa.gov/	Ocean
Communication Buoy	https://www.rolyanbuoys.com/	Product
Underwater Robot	https://www.marketresearchfuture.com/reports/underwater-robotics-market-7605	Market
Magnetic force measuring instrument	https://www.mcmaster.com /products/magnetometers	Product
Ocean	https://www.sofarocean.com /products/spotter	Geography



Dear Greek Marine Rescue Department:

We are writing to you with some ideas aimed at revolutionizing maritime search and rescue operations through the use of advanced predictive modeling, leveraging LSTM models and incorporating significant advancements in computational fluid dynamics. Here are the four major advantages of our model for your reference:

Dynamic Modeling

Developed dynamic predictive model integrating oceanography and physics principles, focusing on seawater density, flow, and topography to forecast mini-submarine trajectories accurately.

Innovative LSTM Integration

Employed LSTM for its proficiency in time series prediction, enhancing the model's ability to forecast future submarine positions based on historical trajectory and ocean physics data.

Enhanced Precision and Versatility

Integrated Kalman Filter for more accurate predictions by utilizing real-time data, while demonstrating adaptability to diverse maritime environments, including the Caribbean Sea.

Scalability and Proven Effectiveness

Modified for scalability, enabling simultaneous tracking of multiple submersibles and achieving high prediction accuracy, as evidenced by validations against historical data and simulations.

These recommendations aim to leverage technological advancements and international cooperation to safeguard lives and assets in maritime explorations. We are committed to continuous improvement and collaboration to enhance safety standards.

Best regards,
Team: *2423654*

Improve Communication

Equip submarines with technology for continuous location and environmental data sharing, reducing uncertainties.

Enhance Crew Preparedness

Implement training simulations based on dynamic environmental conditions, preparing crews for real-world rescue operations.

Global Cooperation

Engage in partnerships for sharing data and technologies, promoting advancements in submarine safety and rescue.

Data Integration

Utilize ocean current and density data to improve predictive models for submarine trajectories.

Versatile Equipment

Prioritize acquisition of sonar, GPS, and ROVs for the mother ship, ensuring readiness for diverse search scenarios.

International Standards

Ensure all submarine operations meet IMO guidelines, focusing on design, operation, and emergency protocols.