Problem Chosen	2024 MCM/ICM Summary Sheet	Team Control Number
B		**2421255**

my title

Summary

my abstract

Keywords: keyword1; keyword2

Contents

1 Introduction

1.1 Background

In the 21st century, there has been a growing interest in the exploration of the ocean in various countries and regions. Maritime Cruises Mini-Submarines (MCMS), a Greek company, specializes in manufacturing submersibles to take people to the deepest parts of the ocean. They have set their sights on leading tourists on exciting adventures to explore sunken shipwrecks in the depths of the Ionian Sea.

To make their submersible operations a reality, we help MCMS to obtain regulatory approval and establish safety protocols to address potential communication loss and mechanical issues, such as propulsion failure. And we develop a predictive model that can track the submersible's location over time. This model considers factors such as sea floor positioning, buoyancy, currents, sea density , and geography.

1.2 Restatement of the problem

Considering the background information and restricted conditions identified in the problem statement,we need to solve the following problems:

- Create a model to predict location of the submersible overtime, figure out the uncertain-ties and find what information can the submersible send back with specific equipment to reduce that.
- Determine additional equipment for searching to carry on the host ship considering the possible costs, and recommend devices for rescuing. on both the host ship and rescue ship if necessary.
- Develop a model using location data to recommend deployment points and search patterns to minimize the time to locate a lost submersible.
- Establish a function that relates the probability of finding the submersible to both time and the accumulated search results.
- Extend the model to cover other tourist destinations and adapt it for multiple submersibles moving in the same general vicinity.

1.3 Our work

to my eq (**??**)

- the angular velocity of the bat,
- the velocity of the ball, and
- the position of impact along the bat.

center of percussion [Brody 1986]

Theorem 1.1. *LaTeX*

Lemma 1.2. *TeX.*

Proof. The proof of theorem. □

2 Assumptions and Justifications

Considering that practical problems always contain many complex factors, first of all, we need to make reasonable assumptions to simplify the model, and each hypothesis is closely followed by its corresponding explanation.

- **Assumptions 1: Rationalize the assumptions for the submersible by considering it as a point mass.**

  When moving in the deep sea, the size and shape of the submersible are negligible compared to the problem we are studying, and the movement of each point on the submersible can be considered to be the same. Therefore, the submersible can be represented by a mass point, following the principle of the center of **mass theorem**.

- **Assumptions 2: Assume that the submarine will no longer provide propulsion.**

  Unless under extreme special circumstances, the submersible pilot will cease all operations and await rescue after experiencing distress. Therefore, it can be inferred that the submersible is powerless on its own once in distress.

- **Assumptions 3: The decision factors only include ocean currents, density, and topography.**

  Based on the principles of fluid mechanics, the motion of the submersible is primarily influenced by water flow and its own dynamics. Ocean currents play a significant role in governing the movement of water bodies in the ocean, and density is associated with the flow of water bodies.so these factors are emphasized and other factors are ignored.

- **Assumptions 4: For the simplicity of the model, decision factors do not interfere with each other.**

3 Notations

The key mathematical notations used in this paper are listed in Table 1

Symbol	Definitions	Units

Table 1: Notations used in this paper

4 Model 1 : Model of submersible position over time

We developed a model to predict how a submersible's position will change over time

4.1 The uncertainties associated with the predictions

There are many factors that will affect the results of model predictions, mainly the following:

- **Ocean current interference**,The influence of ocean currents on the movement of the submersible, which in turn affects the trajectory.
- **Changes in seawater density**,When the temperature and salinity correlation change, it will affect the density of seawater, thereby affecting the buoyancy and resistance. The buoyancy will change the vertical acceleration, and the resistance will change the velocity in the direction of motion.
- **Measure noise and communication delays**, Due to measure noise and communication delays, navigation accuracy is affected,communication efficiency is reduced ,so the results are interfered.
- **Change in the probability of finding the submersible caused by long search and rescue time**, it will lead to difficulty in accumulating data, decreased search efficiency, uncertainty in mission objectives, changes in communication requirements, and impact on location prediction.
- **Neutral buoyancy condition**,Neutral buoyancy conditions cause drift risk, risk of getting wet, depth changes, environmental variations, and changes in forecast results.

4.2 title

4.2.1 Basic assumptions of the model

- Ignore vertical changes in salinity
- The Ionian Sea is small and closed, so horizontal density changes can be ignored.

4.2.2 Temperature, salinity and density at specific depths

In the model, the density of seawater is a crucial factor determining the water resistance and buoyancy of the submarine, which is essential for predicting the trajectory after a submarine accident. Therefore, the relationship between density and submarine depth is first solved.

The relationship between pressure and density, gravitational acceleration and depth is related by this formula.

$$P = \rho g h \tag{1}$$

On the left side of the equation is the pressure at a certain underwater location, while the right side represents the product of seawater density (ρ), local gravity acceleration (g), and diving depth (h). Since the change in depth is significant during the descent, after consulting the data and calculating, It is known that the product of seawater density (ρ) and local gravity acceleration (g) is approximately 0.10045. Therefore, in the formula, it is reasonable to consider P = 0.10045h, therefore we can calculate a certain Pressure corresponding to depth.

By reviewing literature[2] and analyzing the information, we obtained scatter diagrams illustrating the relationship between seawater salinity and pressure, as well as seawater temperature and pressure. This enables us to determine the corresponding temperature and salinity values for a specific depth.

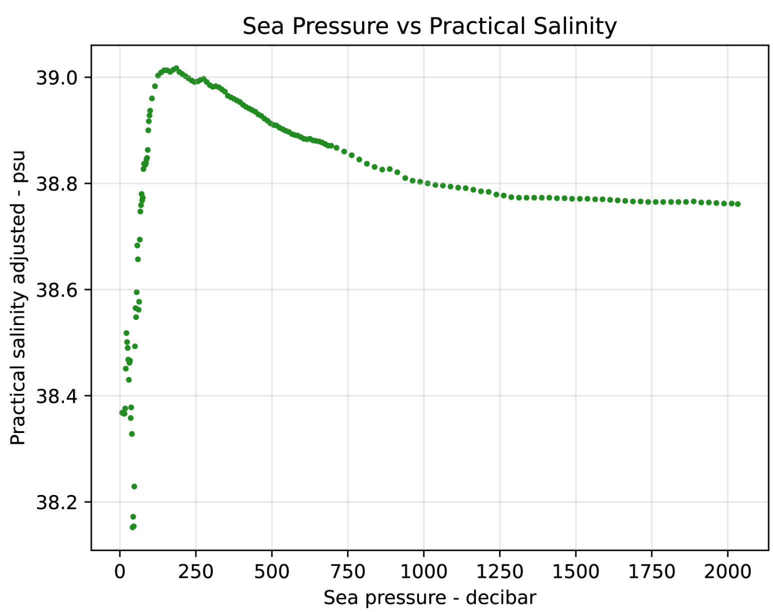

Figure 1: Seawater salinity-pressure scatter point

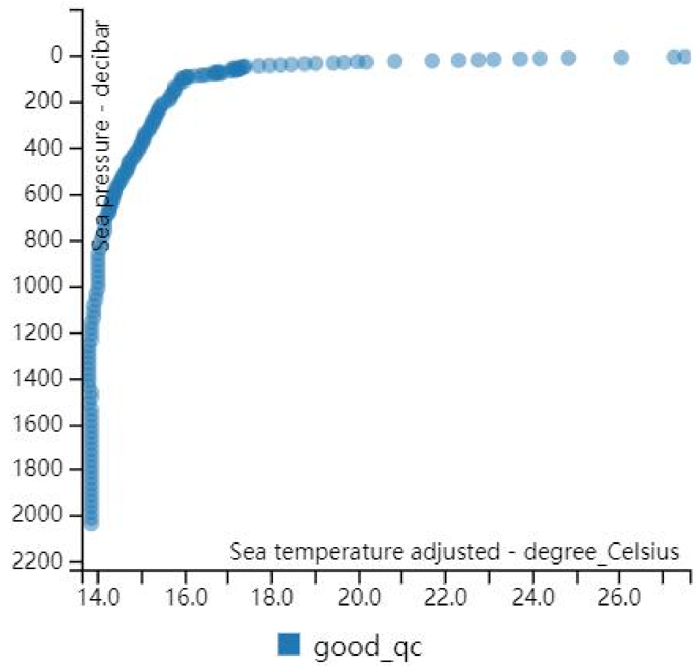

Figure 2: Seawater temperature-pressure scatter plot

After calculating the seawater temperature and salinity, and then considering the relationship between seawater salinity, temperature and seawater density, we can use the Thermodynamic Equation of Seawater to calculate the density of seawater.

$$\rho = \rho_0 + \alpha(T - T_0) + \beta(S - S_0) \tag{2}$$

On the left side of the equation, ρ represents the density at a certain underwater location, ρ_0 is the reference density that is taken as the density of pure water, and $(T - T_0)$ and $(S - S_0)$ denote the temperature and salinity differences from their respective reference values.

From the formula, we can infer that ρ is determined by the three-dimensional linear relationship between T and S. However, because factors such as sea currents and temperature will stabilize after reaching a certain depth, the change in seawater density no longer shows an obvious linear relationship, so we n eed to add a correction formula to calculate more accuratele value of seawater density.

$$\rho = \begin{cases} \rho_0 + \alpha(T - T_0) + \beta(S - S_0) & \text{if } h < 2,000\ m \\ C_1 & \text{if } h \geqslant 2,000\ m \end{cases} \tag{3}$$

By consulting information and data, we calculated that the density of seawater is $2928.717 g/cm^3$ when the depth is more than 2000 meters. So we conclude that the density of seawater is a definite value at a specific depth. Therefore we plotted the density of seawater as a function of depth.

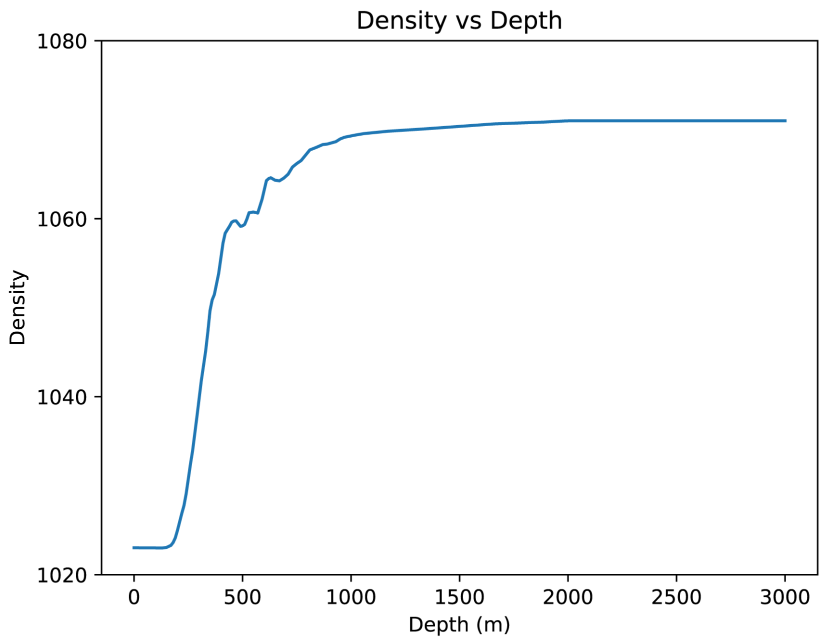

Figure 3: Seawater Density changes with depth

4.2.3 ???

5 Sensitivity Analysis

6 Evaluate of the Mode and Further Discussion

7 Conclusions

8 Strengths and weaknesses

8.1 Strengths

- **Applies widely**
  This system can be used for many types of airplanes, and it also solves the interference during the procedure of the boarding airplane,as described above we can get to the optimization boarding time.We also know that all the service is automate.

- **Improve the quality of the airport service**
  Balancing the cost of the cost and the benefit, it will bring in more convenient for airport and passengers.It also saves many human resources for the airline.

8.2 How to cite?

bibliography cite use [1, 2, 3]

AI cite use [AI: 1, 2, 3]

References

[1] VirtualEXPO GROUP.NauticEXPO.https://www.nauticexpo.com/

[2] Argo (2024). Argo float data and metadata from Global Data Assembly Centre (Argo GDAC) - Snapshot of Argo GDAC of Jannuary 09st 2024. SEANOE. https://doi.org/10.17882/42182#107470 Addison-Wesley Publishing Company, 1986.

[3] Gebco.net. "GEBCO," n.d. https://www.gebco.net/data_and_products/gridded_bathymetry_data/. gebco.net. https://www.gebco.net/data_and_products/gridded_bathymetry_data/GE

[4] CAO Hong-tao, ZHANG Qi-feng, TANG Shi. Current status and key technologies of Remotely Operated Vehicles in international submarine rescue and deep-sea salvage. Ship Science and Technology, 2021, 43(12): 16-20

[5] Zhihui,YAN Kexue,HUANG Rong.Search and rescue helicopter capability requirement analysis[J].Aeronautical Science & Technology,2021,32(01):60-64

Appendices

Appendix A First appendix

Dear, Mr. Alpha Chiang

Sincerely yours,

Your friends

Input matlab source:

```
function [t,seat,aisle]=OI6Sim(n,target,seated)
pab=rand(1,n);
for i=1:n
    if pab(i)<0.4
       aisleTime(i)=0;
    else
        aisleTime(i)=trirnd(3.2,7.1,38.7);
    end
end
```

Appendix B Second appendix

Memorandum

To: MCM office

From: MCM Team 2421255

Subject: MCM

Date: February 4, 2024

This is a memorandum.

some more text **Input C++ source:**

```
//================================================================
// Name        : Sudoku.cpp
// Author      : lst
// Version     : a.0
// Copyright   : Your copyright notice
// Description : Sudoku in C++.
//================================================================

#include <iostream>
#include <cstdlib>
#include <ctime>

using namespace std;

int table[9][9];

int main() {

    for(int i = 0; i < 9; i++){
        table[0][i] = i + 1;
    }

    srand((unsigned int)time(NULL));

    shuffle((int *)&table[0], 9);

    while(!put_line(1))
    {
        shuffle((int *)&table[0], 9);
    }

    for(int x = 0; x < 9; x++){
        for(int y = 0; y < 9; y++){
            cout << table[x][y] << " ";
        }

        cout << endl;
    }

    return 0;
}
```

Report on Use of AI

1. Bing AI
   Query1: write a sort algorithm
   Output: Sure, I can help you with that. Here is a simple implementation of the selection sort algorithm in Python:

```
def selection_sort(arr):
    for i in range(len(arr)):
        min_idx = i
        for j in range(i + 1, len(arr)):
            if arr[min_idx] > arr[j]:
                min_idx = j
        arr[i], arr[min_idx] = arr[min_idx], arr[i]
    return arr
```

   This algorithm sorts an array by repeatedly finding the minimum element from the unsorted part of the array and putting it at the beginning. The time complexity of this algorithm is O(n^2).

   There are many other sorting algorithms available such as bubble sort, insertion sort, merge sort, quick sort, heap sort, and more. Each of these algorithms has its own advantages and disadvantages. You can learn more about them on GeeksforGeeks and Programiz.

   I hope this helps!

2. OpenAl Enie (Nov 5, 2023 version, Ernie 4.0)
   Query1: <insert the exact wording of any subsequent input into the Al tool>
   Output: <insert the complete output from the second query>

3. Github CoPilot (Feb 3, 2024 version)
   Query1: <insert the exact wording you input into the Al tool>
   Output: <insert the complete output fron the Al tool>

4. Google Bard (Feb 2,2024 version)
   Query: <insert the exact wording of your query>
   Output: <irsert the complete output from the Al tool>

5. Bing AI
   Query: write a FFT algorithm
   Output: Sure, here is a Python implementation of the Fast Fourier Transform (FFT) algorithm:

```
import numpy as np

def fft(x):
    n = len(x)
    if n == 1:
        return x
    even = fft(x[0::2])
    odd = fft(x[1::2])
    T = [np.exp(-2j * np.pi * k / n) * odd[k] for k in range(n // 2)]
    return [even[k] + T[k] for k in range(n // 2)] + [even[k] - T[k] for k
      ↪ in range(n // 2)]
```

The FFT is a widely used algorithm for computing the **Discrete Fourier Transform (DFT)** of a sequence. It reduces the number of computations needed from O(n^2) to O(n log n) 12. The basic idea of the FFT is to apply divide and conquer. We divide the coefficient vector of the polynomial into two vectors, recursively compute the DFT for each of them, and combine the results to compute the DFT of the complete polynomial 3.

I hope this helps!