2022

Bayesian Search Study for USW

Szechtman, Roberto; Kress, Moshe

Monterey, California: Naval Postgraduate School

https://hdl.handle.net/10945/71892

NPS-OR-22-006

NAVAL POSTGRADUATE SCHOOL

MONTEREY, CALIFORNIA

BAYESIAN SEARCH STUDY FOR USW

by

Moshe Kress and Roberto Szechtman

October 2022

Distribution Statement A: Approved for public release. Distribution is unlimited.

Prepared for: U.S. Fleet Forces Command. This research is supported by funding from the Naval Postgraduate School, Naval Research Program (PE 0605853N/2098).
NRP Project ID: NPS-22-N208-A

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE

PLEASE DO NOT RETURN YOUR FORM TO THE ABOVE ORGANIZATION.

1. REPORT DATE	**2. REPORT TYPE**	**3. DATES COVERED**	
		START DATE	**END DATE**
10/17/2022	Technical Report	10/24/2021	10/22/2022

3. TITLE AND SUBTITLE
Bayesian Search Study for USW

5a. CONTRACT NUMBER	**5b. GRANT NUMBER**	**5c. PROGRAM ELEMENT NUMBER**
		0605853N/2098
5d. PROJECT NUMBER	**5e. TASK NUMBER**	**5f. WORK UNIT NUMBER**
NPS-22-N208-A; W2223		

6. AUTHOR(S)
Moshe Kress and Roberto Szechtman

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)	**8. PERFORMING ORGANIZATION REPORT NUMBER**
Operations Research Department Naval Postgraduate School 1411 Cunningham Road Monterey, CA 93943	NPS-OR-22-006

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)	**10. SPONSOR/MONITOR'S ACRONYM(S)**	**11. SPONSOR/MONITOR'S REPORT NUMBER(S)**
U.S. Fleet Forces Command	NRP; USFF	NPS-OR-22-006; NPS-22-N208-A

12. DISTRIBUTION/AVAILABILITY STATEMENT
Distribution Statement A: Approved for public release. Distribution is unlimited.

13. SUPPLEMENTARY NOTES

14. ABSTRACT

Adversarial submarine activity in the Atlantic has steadily intensified over the last few years. Furthermore, strategic adversaries have developed sophisticated and stealthy submarines making them much more difficult to locate. The heightened activity coupled with advanced platforms have allowed the United States' adversaries to challenge its dominance in the underwater domain. Though extensive research has been performed on optimized search strategies using Bayesian search methods, most methodologies in the open literature focus on search for stationary objects rather than a search for a moving Red submarine conducted by a Blue submarine. Thusly motivated, we develop a model of an enemy submarine whose goal is to avoid detection. As the search effort is expended, a posterior probability distribution for the enemy submarine's location is calculated based off negative search results. We present a methodology for finding a search pattern that attempts to maximize the probability of detection in a Bayesian framework utilizing Markovian properties. Specifically, we study three different running window methods: a simple network optimization model, a network optimization model that performs updates after every time period planning the entire route, and a dynamic program that only looks two time periods ahead.

15. SUBJECT TERMS

USW, Bayesian search, approximate dynamic programming

16. SECURITY CLASSIFICATION OF:			**17. LIMITATION OF ABSTRACT**	**18. NUMBER OF PAGES**
a. REPORT	**b. ABSTRACT**	**C. THIS PAGE**	UU	41
U	U	U		

19a. NAME OF RESPONSIBLE PERSON	**19b. PHONE NUMBER** *(Include area code)*
Roberto Szechtman	831-656-3311

STANDARD FORM 298 (REV. 5/2020)
Prescribed by ANSI Std. Z39.18

THIS PAGE INTENTIONALLY LEFT BLANK

NAVAL POSTGRADUATE SCHOOL
Monterey, California 93943-5000

Ann E. Rondeau
President

Scott Gartner
Provost

The report entitled "Bayesian Search Study for USW" was prepared for U.S. Fleet Forces Command and funded by the Naval Postgraduate School, Naval Research Program (PE 0605853N/2098).

Distribution Statement A: Approved for public release. Distribution is unlimited.

This report was prepared by:

Moshe Kress
Distinguished Professor

Roberto Szechtman
Professor

Reviewed by:

Dr. Matt Carlyle, Chairman
Operations Research

Released by:

Kevin B. Smith
Vice Provost for Research

THIS PAGE INTENTIONALLY LEFT BLANK

Prepared for: U.S. Fleet Forces Command
This research is supported by funding from the Naval Postgraduate School, Naval Research Program (PE 0605853N/2098)

NRP Project ID: NPS-22-N208-A

Bayesian Search Study for USW

Distribution Statement A

Moshe Kress and Roberto Szechtman
Operations Research Department, NPS

October 20, 2022

Abstract

Adversarial submarine activity in the Atlantic has steadily intensified over the last few years. Furthermore, strategic adversaries have developed sophisticated and stealthy submarines making them much more difficult to locate. The heightened activity coupled with advanced platforms have allowed the United States' adversaries to challenge its dominance in the underwater domain. Though extensive research has been performed on optimized search strategies using Bayesian search methods, most methodologies in the open literature focus on search for stationary objects rather than a search for a moving Red submarine conducted by a Blue submarine. Thusly motivated, we develop a model of an enemy submarine whose goal is to avoid detection. As the search effort is expended, a posterior probability distribution for the enemy submarine's location is calculated based off negative search results. We present a methodology for finding a search pattern that attempts to maximize the probability of detection in a Bayesian framework utilizing Markovian properties. Specifically, we study three different running window methods: a simple network optimization model, a network optimization model that performs updates after every time period planning the entire route, and a dynamic program that only looks two time periods ahead.

1 Background

Submarine activity in the Atlantic within the last few years has steadily intensified, with multiple Red submarines making deployments to the Atlantic Ocean. Additionally, U.S. adversaries are developing highly capable and stealthy submarines equivalent to U.S. Navy (USN) submarine classes. Because of this, U.S. senior leaders have assessed that the Atlantic Ocean is no longer an uncontested battlefield, which has forced the USN to refocus its efforts to challenge such undersea threats. For example, in 2018 the USN reestablished the U.S. Second Fleet to counter adversarial submarine activity in the Atlantic [LaGrone, 2018]. In addition, the Navy recently announced the creation of a new task force of destroyers specifically assigned to be ready on short notice to deploy in response to hunting submarines in the Atlantic [Shelbourne, 2021]. Operationally, crews are receiving extra training and certifications prior to deployments to ensure readiness to face the undersea threat from hostile submarines.

The actions the USN is taking unambiguously illustrate the significance of the threat posed to national security by heightened hostile submarine presence in the Atlantic. Being able to quickly locate and track Red submarines as they deploy to the Atlantic is vital to national security. The Ohio-Class Ballistic Missile Submarines (SSBNs) were designed to be the survivable leg of the nuclear triad. Submarine crews on SSBNs, while on alert, are required to remain undetected so there remains a credible second nuclear strike capability, which offers the President additional flexibility in decision-making and presents a deterrence of nuclear and non-nuclear aggression by strategic adversaries. Hostile submarine activity operating in waters near SSBNs potentially degrades the survivability of the SSBNs, should the former submarines detect and track them. The ability to quickly locate hostile submarines will allow commanders to adjust the location of SSBNs to maintain maximum assurance of their survivability and will help the Navy track the Red submarines to alleviate the threat of their weapons on the homeland.

From a historical perspective, Anti-submarine Warfare (ASW) began in earnest during World War I to counter the Imperial German Navy's strategy of unrestricted submarine warfare [Cares, 2021]. Since then ASW has evolved into two categories: offensive ASW and defensive ASW [Cares, 2021]. In offensive ASW, the goal is to hunt and kill enemy submarines [Cares, 2021]. However it is critical to note that during peacetime operations the goal is modified to locate and maintain contact on the adversarial submarine [Cares, 2021]. On the other hand, the goal in defensive ASW is to defend assets from being attacked from enemy submarines [Cares, 2021]. What is common to both of these efforts is the need to efficiently find enemy submarines. Submarine commanders are given waterspace within which to operate, information regarding a position for an adversary submarine, and possibly intelligence regarding the adversary submarine's mission. With this information, commanders are required to develop a plan to search for the adversary submarine, often over planning horizons (e.g., 12 hours).

2 Literature Review

The earliest search theory traces back to the work done by the United States Navy's Anti-Submarine Warfare Operations Research Group, documented in the classified 1946 report *Search and Screening* [Koopman, 1946], which laid the foundation for search theory [Benkoski et al., 1991]. In 1980, Koopman updated and published an unclassified version of the 1946 report [Koopman, 1980]. The first major problem arising in the literature is the optimal allocation of search effort against a stationary target. Koopman wrote three papers in 1956-1957 that extended the 1946 report that resulted in the determination of the optimal distribution of search effort for a stationary target using an exponential detection function ([Koopman, 1956a], [Koopman, 1956b], and [Koopman, 1957]). [Charnes and Cooper, 1958] developed an algorithm using a convex optimization problem to determine the optimum allocation for search effort. [Richardson and Stone, 1971] utilized Bayesian search methods in the search for the USS Scorpion, whereby a prior distribution for the location of the Scorpion was first developed by assessing nine possible reasons the submarine sank. Experts then assigned credence to each scenario based on plausibility relative to the other scenarios, and Monte-Carlo simulation was used to first select one of the scenarios to then simulate the movement of the submarine according to the selected scenario. The output of the replication was the location (or not) of the submarine. 10,000 replications were performed, creating the prior distribution of the Scorpion. As the search went on, a posterior distribution was calculated based on the negative detections. Similar methods were used by [Stone et al., 2014] in the search for Air France Flight AF 447.

Literature for optimal search against a moving target was limited before 1977 [Benkoski et al., 1991]. According to [Benkoski et al., 1991], the literature for moving targets can be grouped into two areas:

1. articles which address special types of target motion that are amenable to analysis
2. articles which build on an observation made by Brown in 1977 and aim at developing general necessary and sufficient conditions for moving-target problems [that condition with stationary targets] [Benkoski et al., 1991]

In regards to category (ii), the observation made by [Brown, 1980] is that if a time t is chosen and we condition on failure at all times other than t, then the optimal plan maximizes the probability of detection for the stationary problem [Benkoski et al., 1991]. In contrast, in this report, we consider target motion that is amenable to analysis. [Benkoski et al., 1991] argue that there are two special types of target motion in regard to category (i). One deals with targets whose motion is conditionally deterministic, meaning the target's motion depends on stochastic parameters like initial position and velocity, and if the parameters are known, then the target's position will be known [Stone and Richardson, 1974]. [Stone and Richardson, 1974] and [Stone, 1977] study this problem in the case of continuous time. [Iida, 1989] furthers the work of Richardson and Stone by adding the problem of finding the optimal search plan that minimizes the expected search cost minus the expected reward, also known as the expected risk. For scenarios with discrete time and space, [Royset and Sato, 2010] define a convex mixed integer nonlinear program for multiple search assets attempting to detect one or more probabilistically moving targets.

The second category deals with targets whose motion is Markovian, meaning the target moves in a random manner that may be modeled by a Markov process. [Pollock, 1970], [Dobbie, 1974], and [Schweitzer, 1971] study the two cell problem while [Nakai, 1973] investigates a three cell problem and [Kan, 1977] considers n cells. More modern literature involves specialized branch and bound algorithms for finding the optimal path for a single searcher [Foraker, 2011]. This work includes [Washburn, 1998] and [Sato and Royset, 2010]. [Dell et al., 1996] develop a branch-and- bound procedure and six heuristics for solving constrained-path problems associated with multiple searcher moving target problems.

In the next two Sections we first discuss the area in which the Blue and Red submarines operate. Next we describe our modeling approach for Red's movement as a Discrete Time Markov Chain and how we create the prior distribution for Red's location in each time period. We then present a simple network optimization model, the Network Algorithm (without updates) (NA), which lays the foundation for the more complex Network Algorithm with Updates (NAU). Finally, we present our third algorithm, the Dynamic Programming (DP) algorithm that maximizes the probability of finding Red in the next time period, or in the subsequent time period.

3 Red Movement Description

3.1 Grid

We model the possible Red submarine locations in a grid of $n \times n$ cells, each of equal size, which we define as the Search Region (SR). The SR size is an input parameter to the model, and as such it may be expanded or shrunk in order to fit the requisite scenarios as determined by decision makers. For our models, we consider a 200 nautical mile by 200 nautical mile SR with individual cells being 10 nautical miles by 10 nautical miles wide. Figure 1 displays a SR with these parameters.

3.2 Red's Location

We assume the Red submarine starts its movement in one of k possible cells in the SR; each cell is equally likely to be Red's starting position. The set of starting cells is known by Blue forces. We assume that this knowledge is the result of an initial detection from an external sensor to the Blue submarine, possibly the Integrated Undersea Surveillance System (IUSS), which produces a probability distribution for Red's location.

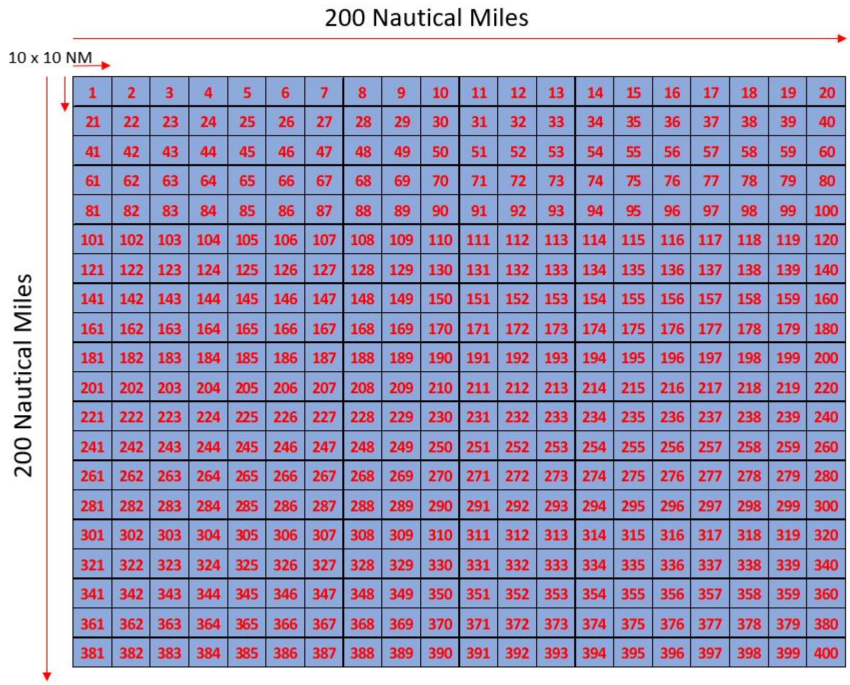

Figure 1: The SR is a 20 x 20 grid, where each cell is 10 nautical miles squared. The northern end of the SR is at the top of the grid.

3.3 Red Movement Model

We model the Red submarine movement as a Discrete Time Markov Chain. A state in the chain is the cell location of the Red submarine, and the transition probabilities govern Red's movement from one cell to another. An additional state is for the possibility that the Red submarine leaves the grid from a cell at the boundary of the SR. Thus the transition matrix is a $(n^2 + 1 \times n^2 + 1)$ matrix. We assume a stationary transition matrix for Red, meaning that the transition probabilities remain constant during the time horizon while the search is being conducted; Red is not affected by Blue's presence. With this methodology we are capable of modeling different mission sets Red may perform simply by changing the transition probabilities.

The transition matrix may have two different interpretations. In the first interpretation Red has a deterministic route but Blue does not know about it for sure, only in probability. The transition matrix represents Blue's belief about Red's movement. As the search unfolds Blue learns more about Red's possible movement. In the second interpretation Red has an overall plan but at each time period it selects the next cell to move into according to the transition matrix. We adopt the second interpretation in this report and leave the first interpretation to future research.

We often assume a scenario in which Red is nearing its endurance limit and is on a northerly track to return to its home port. For this scenario, in each time step the Red submarine either stays in its same position or moves to one of the five adjacent cells to the north, north east, north west, east, and west. Figure 2 illustrates how Red may move from the center cell. If the Red submarine is located in a cell at the boundary of the SR, we assume it either remains in the same location, moves between the adjacent cells in the SR that are to the north, north east, north west, east, and west, or it leaves the SR. If Red leaves the SR, we assume that it never returns to the SR and the search effort will be a failure. Table 1 is an example of a transition matrix that illustrates this behavior for the case where *n=3* as the case for *n=20* results in a matrix of size 401 by 401. This condensed transition matrix is similar to the matrix for the *n=20* case; the number of rows and columns are increased from 10 to 21. Inherently, this means that we assume Red is transiting at 10 knots since the grid size is 10 nautical miles wide and Red is limited to move at most one cell in each time period. This is without loss to generality since the cells' size may be redefined to make this happen. The probability the Red submarine moves to each possible cell is assumed to be equal amongst the set of possible moves. However, in the cases where the Red submarine is in a boundary cell where it can exit the SR by moving in more than one direction, we neglect the different ways in which the Red submarine can exit the SR and group them to be one possible move.

We argue that with changes to the transition probabilities between cells in the transition matrix we may vary the mission set Red is assumed to be conducting. For example, if it is believed that Red is transiting back to its home port in a northerly direction, the transition matrix will only allow for position changes between northerly cells. Another scenario is that Red is performing a barrier search north to south, so we only allow transitioning between states to the north or south.

To update the SR with the probability of Red's location, we define, for each time period t, an n^2 dimensional vector λ_t of probabilities, where each entry is the current probability of the Red submarine being in the corresponding cell. We also reduce the transition matrix to size $(n^2 \times n^2)$ by removing the row and column corresponding to the state where the Red submarine exits the SR. This is done because we do not maintain the probability that Red is outside of the SR. The resulting transition matrix is defined as M, where the M_{ij} entry is equal to the probability of Red transitioning from cell i to j.

$$M = \begin{bmatrix} M_{11} & M_{12} & \cdots & M_{1n} \\ M_{21} & M_{22} & \cdots & M_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ M_{n1} & M_{n2} & \cdots & M_{nn} \end{bmatrix}$$

Then, for all remaining time periods, we perform the calculation:

$$\lambda_{t+1} = \lambda_t \times M \tag{1}$$

to obtain the target's distribution at time $t+1$. The resulting matrix sized $n \times n$ is the projected probabilities of Red's location at time t.

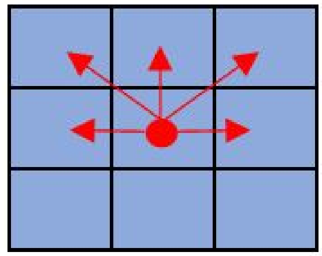

Figure 2: A 3 by 3 SR where Red may only move to the northwest, north, northeast, west, east, or the same cell.

Table 1: 9 Cell SR Transition Matrix

State	1	2	3	4	5	6	7	8	9	Leave
1	1/3	1/3	0	0	0	0	0	0	0	1/3
2	1/4	1/4	1/4	0	0	0	0	0	0	1/4
3	0	1/3	1/3	0	0	0	0	0	0	1/3
4	1/5	1/5	0	1/5	1/5	0	0	0	0	1/5
5	1/6	1/6	1/6	1/6	1/6	1/6	0	0	0	0
6	0	1/5	1/5	0	1/5	1/5	0	0	0	1/5
7	0	0	0	1/5	1/5	0	1/5	1/5	0	1/5
8	0	0	0	1/7	1/7	1/7	1/7	1/7	1/7	1/7
9	0	0	0	0	1/5	1/5	0	1/5	1/5	1/5
Leave	0	0	0	0	0	0	0	0	0	1

4 Blue Movement Models

In this section, we discuss three different algorithms. The first is a network optimization algorithm that minimizes the probability of not detecting Red, based on the λ_t values described above. In this algorithm, we find the route that minimizes the probability of not detecting Red over the time horizon. Next, we present a similar network algorithm; however, this algorithm performs Bayesian updates each time period of the distribution of Red's position resulting from Red not being detected in each time period and then recomputes Blue's route. Finally, we present a Two-Step Dynamic Programming algorithm that, at time step t, aims to maximize the probability of finding Red in either the *t+1* time step, or if not then in the *t+2* time step after the Bayesian update of the distribution of Red's position resulting from Red not being detected in the $t + 1$ time step. It is important to note that all three algorithms generate the search plans prior to Blue expending any physical search effort.

4.1 Network Algorithm (without updates) (NA)

The Blue search asset starts the search in one "border cell" B which is a subset of the cells in the SR that lies on one of the edges of the SR. Blue has T time periods to find the Red submarine and may move only one cell each time period. The objective function for the NA is to minimize the probability of not detecting Red over the time horizon T, which is equivalent to maximizing the probability of detecting Red during the time horizon.

4.1.1 Assumptions

We assume that Blue forces have intelligence regarding the starting position of Red and its general mission. This information is manifested in knowing Red's transition matrix and the k cells from which Red may initially start as well as Red's initial location probability distribution; Red is equally likely to start from any of the k cells. We also assume that Blue search asset has perfect sensors, meaning there are no false positive or false negative detections; the sensors are not affected by the environment; and that Blue detects the Red submarine if they both lie in the same cell at the same time. Blue cannot detect Red if they are in different cells. We assume that Blue may only move to a neighboring cell each time step, which inherently means that Blue and Red travel at the same speeds since both may only move one cell in each time period and Blue's search takes no time.

4.1.2 Notation

- Cell: $i \in I$, i = 1,...,I set of all possible cells
- $B \subset I$ set of "border cells" from which Blue may start the search.
- P_i^t, $i \in I$, t = 1,...,T: Probability Red is in cell i during period t
- Accessibility parameter:

$$a_{i,j} = \begin{cases} 1 & \text{If Blue can move from cell } i \in I \text{ to cell } j \in I \text{ in one time period} \\ 0 & \text{Otherwise} \end{cases} \tag{2}$$

4.1.3 Decision Variables

$$x_{i,j}^t = \begin{cases} 1 & \text{If Blue moves from cell } i \in I \text{ to cell } j \in I \text{ in time period } t \in 2, 3..., T \\ 0 & \text{Otherwise} \end{cases} \tag{3}$$

$$x_{0,i}^1 = \begin{cases} 1 & \text{If Blue starts the search in cell } i \in B \\ 0 & \text{Otherwise} \end{cases} \tag{4}$$

4.1.4 Objective Function

The goal is to maximize the probability of detecting the Red submarine over the planning horizon, T, or similarly, to minimize the probability of not detecting the Red submarine during that time. Thus, our objective function is:

$$\min \prod_{i \in B} (1 - P_i^1)^{x_{0,i}^1} \prod_{t=2}^{T} \prod_{j \in I} (1 - P_j^t)^{\sum_{i \in I} a_{i,j} x_{i,j}^t} \tag{5}$$

Taking the log of the equation 5 yields:

$$\min \sum_{i \in B} \log(1 - P_i^1) x_{0,i}^1 + \sum_{t=2}^{T} \sum_{j \in I} \log(1 - P_j^t) \sum_{i \in I} a_{i,j} x_{i,j}^t \tag{6}$$

By taking the log of our objective function, this allows us to maintain a linear objective function.

4.1.5 Constraints

$$\sum_{j \in I} a_{i,j} x_{i,j}^t - \sum_{k \in I} a_{k,i} x_{k,i}^{t-1} = 0 \ \forall i \in I,\ \forall t = 3, 4, ..., T \tag{7}$$

$$\sum_{j \in I} a_{i,j} x_{i,j}^2 - x_{0,i}^1 = 0 \ \forall j \in B \tag{8}$$

$$\sum_{i \in B} x_{0,i}^1 = 1 \tag{9}$$

The constraints above are common flow balance constraints seen in network problems. The first, equation 7, ensures that Blue may only search one cell in the next time period that is accessible to it for time periods 3 and on. Equation 8 ensures the same; however, it does this for the second time period. We must have a separate constraint here because we use two different sets of decision variables for Blue's location: one set for the first time period where Blue begins its search route and another set for all remaining time periods. This constraint ensures that in time period 2, we only pick one cell that is accessible to Blue from the cell Blue picked in the first time period. Finally, equation 9 ensures that Blue may only pick one cell in the first time period.

4.2 Network Algorithm with Updates (NAU)

The algorithm in Section 4.1 is limited because it obtains the optimal search plan with only the knowledge we had prior to conducting the search. Each time period, we learn something new. From a negative search, we learn that Red is not in the cell we searched, which is information that we can use. Next, we reformulate the algorithm presented in Section 4.1 by adding an update each time period. We assume that each cell that Blue visits does not contain Red. This assumption is intuitive because if we find Red during our search, our search is done. So, to make a search plan over the remaining planning horizon, each time period we assume a negative search result. Following this assumption, we use Bayes Theorem to update the current time period's probability distribution of the Red submarine location in the SR. Using this updated distribution along with Red's stationary transition matrix, we calculate the remaining time period's posterior distributions of Red's location. This update is discussed in full detail in a later section. These repeated updates lead to an algorithm comprising a sequence of optimization problems to find the best cells for Blue to search prior to any search being conducted.

4.2.1 First Optimization Problem

The first optimization problem is exactly the same as the NA algorithm presented in Section 4.1. The first optimization problem finds the optimal route, given the currently estimated location probabilities in the SR without performing any updates. Once we have our initial optimal route and begin visiting cells, we perform updates on the distribution of Red's location due to negative search results and solve new optimization problems.

4.2.2 Follow On Optimization Problem

Once the first optimized plan is determined, the Blue search asset conducts the search. To reiterate an important assumption for this algorithm, we assume that Blue has a perfect sensor, meaning that if Blue searches a cell, there is a 100% probability that it will detect Red if Red is located in that cell at the same time. To perform the update to the search resulting from a negative search, we apply the following process:

1. Since Blue starts at an edge of the SR and Red may start in any part of the SR, including the extreme other end of the SR, there may be several time periods where it is not possible for Red to travel to a cell in which Blue may also have reached in the same time. Therefore, we find the first time period, t^* where it is possible for Blue and Red to be in the same cell. Based on the original search plan, we find the cell in which Blue will be located at in time period t^*, denoted c^*.

2. Determine the probability Red is in c^*, and set this probability equal to 0 since we assume we did not find Red in this cell.

3. Add the probability Red was in c^* proportionally to all the other cells in the grid.

4. Recompute the remaining time periods probability distribution for Red's location in the SR using the updated probability distribution computed in steps 1 and 2 for the current time period and Red's stationary transition matrix.

5. Re-solve the optimization problem, starting at the $t^* + 1$ time period.

6. Using the new search plan just computed, determine the next cell to move and replace c^* with this cell.

7. Re-perform steps 2 through 6.

Steps 2 through 6 are re-performed for all time periods up until *T-1*. We do not perform any further updating in the *Tth* time period because we assume that we are done searching after we complete the search in this time period.

The optimization algorithms needs to be slightly adjusted to account for having a known cell we are starting from. The new formulation is as follows:

Notation

- Cell notation: $i \in I, i = 1, ..., I$ set of all possible cells
- t^*: first time period it is possible for Red and Blue to be in the same area simultaneously
- c^*: cell searched by Blue in this time period
- P_i^t, $i \in I, t = t^* + 1, ..., T$: Probability Red is in cell i during period t
- Time period notation: $t \in T, t = t^* + 1, ..., T$, set of all remaining time periods
- Accessibility parameter:

$$a_{i,j} = \begin{cases} 1 & \text{If Blue can move from cell } i \in I \text{ to cell } j \in I \text{ in one time period} \\ 0 & \text{Otherwise} \end{cases} \tag{10}$$

Decision Variables

$$x_{i,j}^t = \begin{cases} 1 & \text{If Blue moves from cell } i \in I \text{ to cell } j \in I \text{ in time period } t \in t^* + 1, ..., T \\ 0 & \text{Otherwise} \end{cases} \tag{11}$$

Objective Function

$$\min \sum_{t \in T} \sum_{j \in I} \log(1 - P_j^t) \sum_{i \in I} a_{i,j} x_{i,j}^t \tag{12}$$

Constraints

$$\sum_{j \in I} a_{c^*,j} x_{c^*,j}^{t^*+1} = 1 \tag{13}$$

$$\sum_{j \in I} a_{i,j} x_{i,j}^t - \sum_{k \in I} a_{k,i} x_{k,i}^{t-1} = 0 \; \forall i \in I, \; \forall t = t^* + 2, ..., T \tag{14}$$

$$\sum_{i \in I} \sum_{j \in I} x_{i,j}^t = 1 \; \forall t \in t^* + 1, ..., T \tag{15}$$

Again, we see common network constraints. Equation 13 ensures Blue can only move to one cell accessible to the cell just searched. Equation 14 ensures for all remaining time periods Blue can only move one cell in the next time period accessible to it currently. Finally, the constraint in equation 15 ensure each time period only one cell is searched. This constraint needs to be added because equation 13 only covers the cells accessible from Blue's current position. Without this constraint, other cells in time period $t^* + 1$ would be searched.

4.3 Two Step Dynamic Programming (DP) Algorithm

In the Two Step DP Algorithm, Blue chooses cell $j \in I$ to maximize the probability of finding Red in the next step or, failing that, in the step after. That is, given the current period is t, Blue chooses the cell that maximizes the probability of finding Red in the next time step ($t+1$) or given that Blue did not find Red in the $t+1$ time step, Blue maximizes finding Red in the $t+2$ time step. This algorithm performs the Bayesian update that was also performed in the NAU algorithm presented in Section 4.2. For each possible cell that Blue may move to in the $t+1$ time step, we assume a negative detection and update the posterior distribution for Red in the $t+2$ time step in order to determine the cell with the highest probability of detecting Red in the next two time steps.

4.3.1 Notation

- t^*: First time period where it is possible for Red and Blue to be in the same area simultaneously
- r^*: First row of cells in the SR where it is possible for Red and Blue to be simultaneously
- c^*: Cell in r^* with the highest probability Red is located within it
- $\mathcal{A}(i)$: set of positions accessible to Blue from position i in one step
- P_i^{t+1}: Probability Red is in cell i in period $t+1$
- P_i^{*t+2}: Probability Red is in cell i in period $t+2$ after the Bayesian update has been performed.

4.3.2 Formulation

$$j \in \arg \max_{j \in r^*} \left\{ P_j^{t^*} \right\}, \text{for } t = t^* \tag{16}$$

$$j \in \arg \max_{j \in \mathcal{A}(i)} \left\{ P_j^{t+1} + (1 - P_j^{t+1}) \max_{k \in \mathcal{A}(j)} P_k^{*(t+2)} \right\}, \forall t \in t^* < t \leq T-2 \tag{17}$$

$$j \in \arg \max_{j \in \mathcal{A}(i)} \left\{ P_j^{t+1} \right\}, \text{for } t = T-1 \tag{18}$$

4.3.3 Explanation

In this algorithm, we find the first time period where it is possible for both Red and Blue to be in the same cell (t^*). Since in our scenarios Blue starts from the north and travels south while Red starts in the south and travels north, there will be a single row of cells in the SR where both Red and Blue may be after t^* time periods. This row is denoted r^*, and as equation 16 specifies, we pick the cell with the highest probability to start the search route (denoted c^*), which is a purely greedy approach. Then, as we have done in the past, we assume a negative detection and compute the posterior distributions for Red's location for the remaining time steps. The computation of the posterior distributions is similar to how they were computed previously in the NAU algorithm of section 4.2. Following this, we then use equation 17 to find the next cell to search. We find the cell accessible from c^* in one time step that maximizes the probability of finding Red in the next time step, or given we do not find Red then, maximizes the probability of finding Red in the time step after. To do this, for each of the cells accessible from c^*, we compute posterior distributions for Red's location for the *t+2* time step assuming Red was not in the cell chosen in the *t+1* time step. For example, if there are six cells accessible from c^*, then this will result in six different posterior distributions computed for the *t+2* time step. From these posterior distributions, we determine the cell accessible from the cell chosen in the *t+1* time step with the maximum probability Red is in it. We take this probability and multiply it by the probability Red is not in the cell chosen in the *t+1* time step.

A similar process is repeated for all remaining time periods up until the *T-1* time period. Since this algorithm looks two time periods ahead, we cannot perform this computation in the *T-1* time period, because we only have one time period remaining. Thus, in this time period, we again apply a greedy approach specified in equation 18. The search path is computed prior to any search being conducted.

5 Results

So far we presented three different approaches to determining an optimal route for a single searcher against a moving target in a SR. The first, the simple NA algorithm is used for the first optimization problem in our second algorithm, the NAU, which comprises several iterations of the NA with Bayesian updates in the probability distributions for Red's movement. The third algorithm is the DP algorithm. All three algorithms attempt to find a search plan by minimizing the probability of not detecting Red during the search, or similarly by maximizing the probability of detecting the Red submarine. However, how each goes about solving this problem is different. In the NA algorithm, the goal is to maximize the probability of detection over the entire planning horizon. In the NAU algorithm, the goal is also to maximize the probability of detection over the entire planning horizon, but as search effort is expended and information regarding empty cells is obtained, updates are performed on the posterior distribution for Red's location and follow-on optimization algorithms are created for each remaining time period. In the DP algorithm, in each time period, we only look two steps ahead. The goal at time period t is to maximize the probability of detecting Red in the next time period (*t+1*), or given Red is not found in the *t+1* time period, we maximize the probability of finding it in the *t+2* time period. The fundamental difference between the two network algorithms and the DP algorithm is the time horizon for which we are planning the search: in the two network algorithms we plan over the entire remaining time horizon whereas in the DP algorithm we only look two steps ahead. A fundamental difference between the NAU and the DP algorithms is when the Bayesian updates are applied. In the NAU algorithm, the update is applied once each time period to update the probability distribution for Red's location to account for Red not being in the cell that was directed to be searched in the current time period. In the DP algorithm, the Bayesian updates are applied twice: once to determine which cell will maximize finding Red in two time steps assuming Red was not found in the next time step and then a second time in a manner similar to the

NAU algorithm. In all algorithms, the search plans are generated prior to conducting any physical search effort by Blue.

In this section, we present different scenarios to compare the three algorithms: NA, NAU, and DP. For each scenario, we simulate the "actual" movement of Red using the initial probability distribution for Red's location and its transition matrix, which generates Red's route. We generate 10,000 different routes for Red. Each of the three algorithms are used to determine Blue's search plan for the scenario. In each scenario, we assume Blue was pre-positioned to be in the northern end of the SR, so Blue begins the search in the northernmost row of the SR, and Blue only has 20 time periods to find Red. The 20 time period planning horizon is chosen to give approximately 10 to 16 hours of dedicated search once the transit time is incorporated. Since within each scenario the transition matrix and set of possible starting cells are the same, the Blue search plans are generated only once for each algorithm. We compare the 10,000 different routes for Red to the search plans generated from the three algorithms to determine if Red and Blue are in the same cell during the same time period, which corresponds to a detection. The proportion of replications in which there is a detection will be our measure of efficiency when comparing the algorithms. This proportion is an estimate of the probability of detection. We calculate 95% confidence intervals for each probability of detection using the normal approximation. For each scenario and each algorithm, we also compute empirical Cumulative Distribution Function (CDF) for the probability of detecting Red during or before each time period. The CDFs are determined by tabulating the number of replications Red was detected by Blue during or before each time period and dividing by the total number of replications (10,000).

The scenarios vary the cell from which Red may start as well as varying the number of possible cells from which Red is believed to start by Blue. We will also vary Red's stationary transition matrix, which dictates Red's movement, to fit different mission sets. However, Red's stationary transition matrix is constant within a scenario. Figure 3 illustrates the different scenarios regarding Red's possible starting cell(s). Blue assumes each cell in the collection of Red's initial position cells is equally likely. In a final scenario, we illustrate how better intelligence, giving less uncertainty regarding Red's movement, affects the probability of detection. In all scenarios considered, Blue starts the search in any of the cells in the northernmost row of the SR.

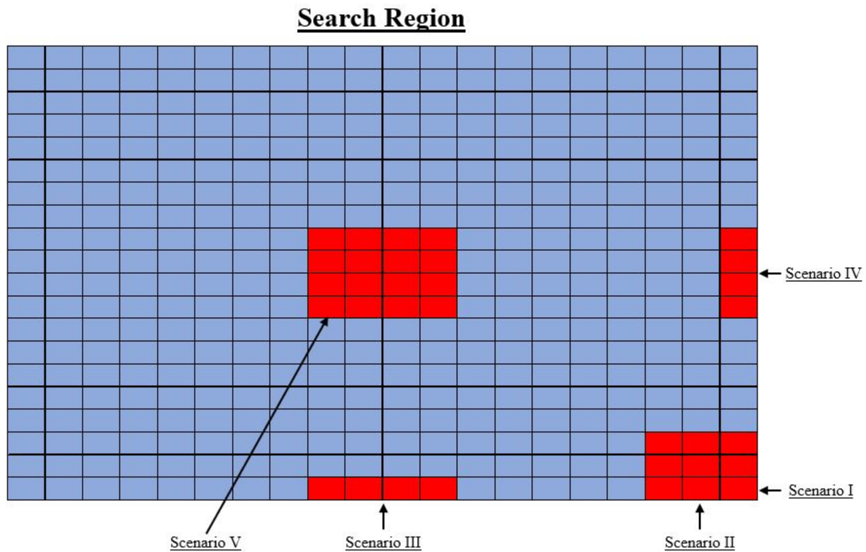

Figure 3: In Scenarios I-V, we vary the starting position for Red and the number of cells from which Red may start. Note that for Scenario I, the starting position is the bottom right cell of the SR. The northern end of the SR is at the top of the grid. The top left cell is (1,1), and the bottom right cell is (20,20).

5.1 Scenario I

In the first scenario we consider a basic case where Red's initial location is known (with probability 1), based on exogenous information obtained from intelligence sources, in only one cell in the SR, the cell that is furthest south and east in the SR (see Figure 3, cell (20,20)). There is also intelligence that Red has reached the end of its endurance limit and is returning to its home base, so it moves along a northerly track (previously discussed and illustrated by Figure 2). The Blue search asset is north of the SR to start the search, so it may start in any of the cells in the northernmost row of the SR.

5.1.1 NA and NAU Results

Table 2 summarizes the search plans generated by the NA and NAU algorithms used in all the replications. In the table, the rows correspond to time periods. The first column corresponds to the NA search plan, and the remaining columns are the NAU search plans updated after search effort is expended in the time period listed for the column. The entries are the (x,y) coordinates in the grid. This representation applies to all other scenarios; however, as we will discuss we will only present a reduced output for the remaining scenarios.

Since Red starts in the bottom right of the SR and Blue starts anywhere in the northernmost row of the SR in this scenario, there are several time periods where it is not possible for Red and Blue to be in the same cell. Therefore, until the probability that Red and Blue may be in the same cell is positive, we do not compute an updated search plan as there is nothing to update yet. The first time period it is possible for the two vessels to be in the same area is in time period 10, so we begin updating then.

Time Period	NA	10	11	12	13	14	15	16	17	18	19
1	(1,15)	-	-	-	-	-	-	-	-	-	-
2	(2,15)	-	-	-	-	-	-	-	-	-	-
3	(3,16)	-	-	-	-	-	-	-	-	-	-
4	(4,17)	-	-	-	-	-	-	-	-	-	-
5	(5,18)	-	-	-	-	-	-	-	-	-	-
6	(6,18)	-	-	-	-	-	-	-	-	-	-
7	(7,19)	-	-	-	-	-	-	-	-	-	-
8	(8,19)	-	-	-	-	-	-	-	-	-	-
9	(9,18)	-	-	-	-	-	-	-	-	-	-
10	(10,19)	-	-	-	-	-	-	-	-	-	-
11	(11,19)	**(11,19)**	-	-	-	-	-	-	-	-	-
12	(12,18)	(12,18)	**(12,18)**	-	-	-	-	-	-	-	-
13	(13,18)	(13,18)	(13,18)	**(13,18)**	-	-	-	-	-	-	-
14	(13,19)	(13,19)	(13,19)	(13,19)	**(14,18)**	-	-	-	-	-	-
15	(13,18)	(13,18)	(13,18)	(13,18)	(13,18)	**(13,18)**	-	-	-	-	-
16	(12,18)	(12,18)	(12,18)	(12,18)	(12,19)	(12,18)	**(12,18)**	-	-	-	-
17	(11,18)	(11,18)	(11,18)	(11,18)	(12,18)	(11,18)	(11,18)	**(11,18)**	-	-	-
18	(11,17)	(11,17)	(11,17)	(11,17)	(11,18)	(11,17)	(10,18)	(10,17)	**(12,17)**	-	-
19	(11,18)	(11,18)	(11,18)	(11,18)	(11,17)	(11,18)	(10,17)	(10,18)	(11,17)	**(11,17)**	-
20	(10,18)	(10,18)	(10,18)	(10,18)	(10,18)	(10,18)	(10,18)	(10,17)	(10,17)	(10,16)	**(10,16)**

Table 2: The column with heading NA represents the (x,y) coordinate in the SR where Blue is directed to search by the NA algorithm. The remaining columns are the cells to be searched after incorporating the Bayesian update and re-running the optimization model. The headings for these columns are the time periods during which the plans were computed. The cells in bold are the actual cells to be searched by Blue and a part of the NAU search plan.

As previously discussed, the original search plan listed is the plan made without including updates for not finding Red and uses only the distribution of Red's initial position and its transition matrix. In each time period Blue expends search effort, Blue learns where Red is not (assuming Blue does not find Red). Thus, we hope that as we update the posterior distributions for Red's location we may come up with a better search plan. The NAU algorithm assumes that Red will not be found in each searched cell and updates its search plan with this information. The updated search plan directs Blue where to search for all remaining time periods. However, since we are computing a new search plan after each time period, we only need to concern ourselves with the first cell our new search plan tells us to search (the cells in bold in Table 2) because after we search this cell, we will compute a new search plan using the information that Red is not found. An important observation is to compare the first cell to search in the updated plans to the cell from the original search plan before any updates were performed. This can give us a sense of how accurate the original search plan is.

Because it is not possible for Blue and Red to be in the same area until time period 10, the cells Blue visits in the first 10 time periods are not cells Blue truly searches since we know that Red will not be there. So, we only need to concern ourselves for time periods 10 and on. Additionally, since after each period we compute a new search plan which assumes Red has not been found, we only need to concern ourselves with the first cell to search for each new search plan. Therefore, we define an *effective* search plan for the NAU algorithm to be the first cell to search from the original search plan where there is positive probability Red and Blue may both be in the same cell plus the cells in bold in Table 2, which correspond to the first cell to search after an update is performed with the information that Red is not found. Table 3 displays this *effective* search plan.

Table 3: Scenario I NAU Effective Search Plan

Time Period	
10	(10,19)
11	(11,19)
12	(12,18)
13	(13,18)
14	(14,18)
15	(13,18)
16	(12,18)
17	(11,18)
18	(12,17)
19	(11,17)
20	(10,16)

From the 10,000 paths simulated for Red, Red was detected 849 times from the search plan generated by the NA algorithm and 876 times from the NAU algorithm, which correspond to (0.0794,0.0904) and (0.0821,0.0931) 95% confidence intervals for the probability of detection, respectively.

5.1.2 DP Results

Table 4 summarizes the search plan produced as a result of the DP algorithm. Again, since it is only possible to detect Red in time periods 10 and on, we present only these results.

Of the 10,000 different paths simulated for Red, Red is detected using this search plan 878 times corresponding to the 95% confidence interval of (0.0823,0.0933) for the probability of detection.

Table 4: Scenario I DP Search Plan

Time Period	
10	(10,19)
11	(11,19)
12	(12,18)
13	(13,18)
14	(14,18)
15	(13,18)
16	(12,18)
17	(11,18)
18	(12,17)
19	(11,17)
20	(12,18)

5.1.3 Discussion

Between the two network algorithms, we see improved performance from the updates we perform using the NAU algorithm. Also, we observe that the effective search plan produced by the NAU and the DP algorithms yield similar plans. In fact, only in the very last time period do we see different cells to search. However, the DP algorithm slightly outperforms the NAU algorithm for this scenario, so the last cell produced from the DP algorithm more often results in a detection. It is important to note, though, that there is overlap between the confidence intervals, meaning that performance is close between the three methods.

Figure 4 presents an empirical CDF for the probability of detecting Red over the time horizon with corresponding 95% confidence bands. We can see that the NAU and DP algorithms perform quite closely, with the DP algorithm slightly outperforming the NAU algorithm in the very last time period.

A possible explanation for the low percentage of detections is the fact that Red lies near the edge of the SR; therefore, there are many opportunities for Red to leave the SR never to return before it is detected. In fact, the empirical probability that Red leaves the SR during or before time period 10 is 57%. As we vary Red's starting position to be in different positions where it is less likely for Red to leave the SR, we expect better empirical probabilities of detection.

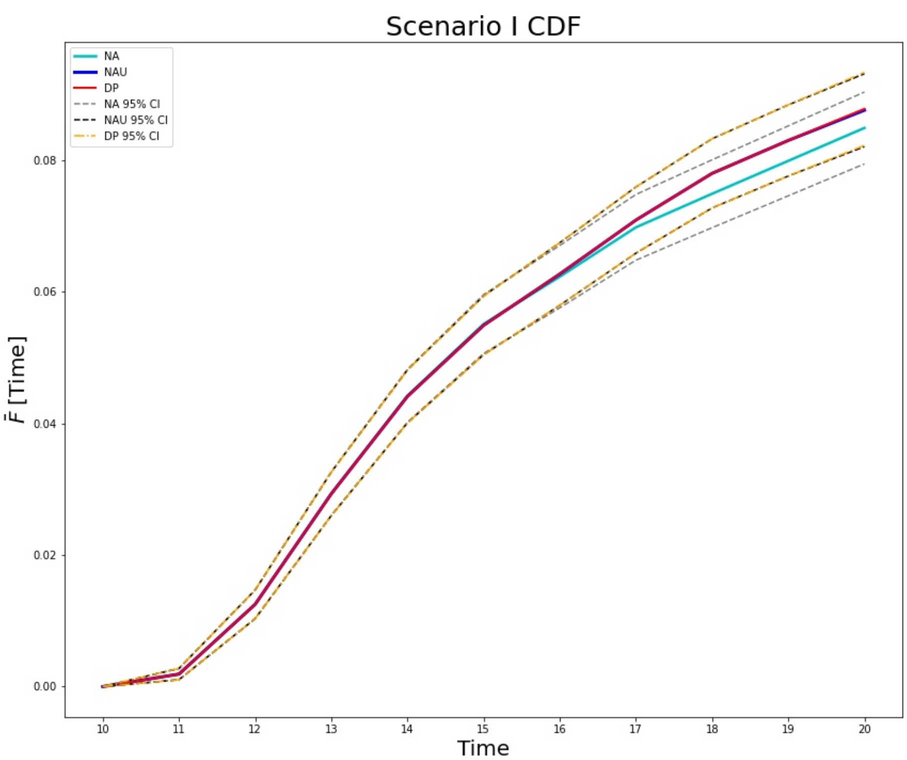

Figure 4: Empirical CDF for Scenario I. The CDF shows the empirical probability of finding Red in or before the time period on the x-axis. Since it takes 10 time periods for Blue and Red to be in the same area, the probability of finding Red before time period 10 is 0.

5.2 Scenario II

In our next scenario we consider the case where Red is detected in the bottom right (southeast corner) of the SR; however, this time Red may be in a 3×3 cluster of cells distributed uniformly (see Figure 3, cells (18,18), (18,19), (18,20), (19,18), (19,19), (19,20), (20,18), (20,19), (20,20)), meaning the probability Red starts in any of the nine cells is 1/9. Thus, when beginning Red's route, we first sample from a uniform distribution to determine the initial starting cell for Red. We again use the same mission set and resulting transition matrix discussed before, and Blue once again starts in any of the cells in the northernmost row of the SR.

5.2.1 NA and NAU Results

Table 5 displays the search plans for the NA and NAU algorithms in all replications. In this scenario, the first time period where there is positive probability Red and Blue may be in the same cell is in time period 9; therefore, we present only the plan for time period 9 on.

Table 5: Scenario II NA and NAU Search Plans

Time Period	NA	NAU
9	(9,18)	(9,18)
10	(10,18)	(10,18)
11	(11,18)	(11,18)
12	(12,18)	(12,18)
13	(13,18)	(13,18)
14	(12,18)	(12,18)
15	(11,18)	(13,18)
16	(11,17)	(12,18)
17	(11,18)	(11,17)
18	(10,18)	(10,17)
19	(9,18)	(9,18)
20	(9,17)	(8,18)

Using these search plans and 10,000 simulations for Red's route, Red is detected 1,150 and 1,173 times for the NA and NAU algorithms, respectively with the corresponding 95% confidence intervals for the probability of detection of (0.1087,0.1213) and (0.1110,0.1236).

5.2.2 DP Results

Table 6 summarizes the search plan used for all the replications. This search plan resulted in 1,173 times where Red was detected, which corresponds to a 95% confidence intervals for the probability of detection of (0.1110,0.1236).

Table 6: Scenario II DP Search Plan

Time Period	
9	(9,18)
10	(10,18)
11	(11,18)
12	(12,18)
13	(13,18)
14	(12,18)
15	(13,18)
16	(12,18)
17	(11,17)
18	(10,17)
19	(9,18)
20	(8,18)

5.2.3 Discussion

Between the two network algorithms, we once again see a slight improvement using the NAU algorithm. Also, we again observe similar search plans produced from NAU and DP algorithms. In this scenario, the algorithms resulted in exactly the same search plan!

We do see an improvement over the 8-9% probability of detection of the previous scenario; this is to be expected because there are now more cells that are further inside the SR for Red's initial position and less opportunities for Red to leave the SR. The empirical probability for Red leaving the SR before time period 9 is 39%.

Figure 5 presents the empirical CDF for this scenario. The 95% confidence intervals for the probability of detection once again have overlap between the algorithms. The NAU and DP algorithms have the same confidence band since they are the same search pattern.

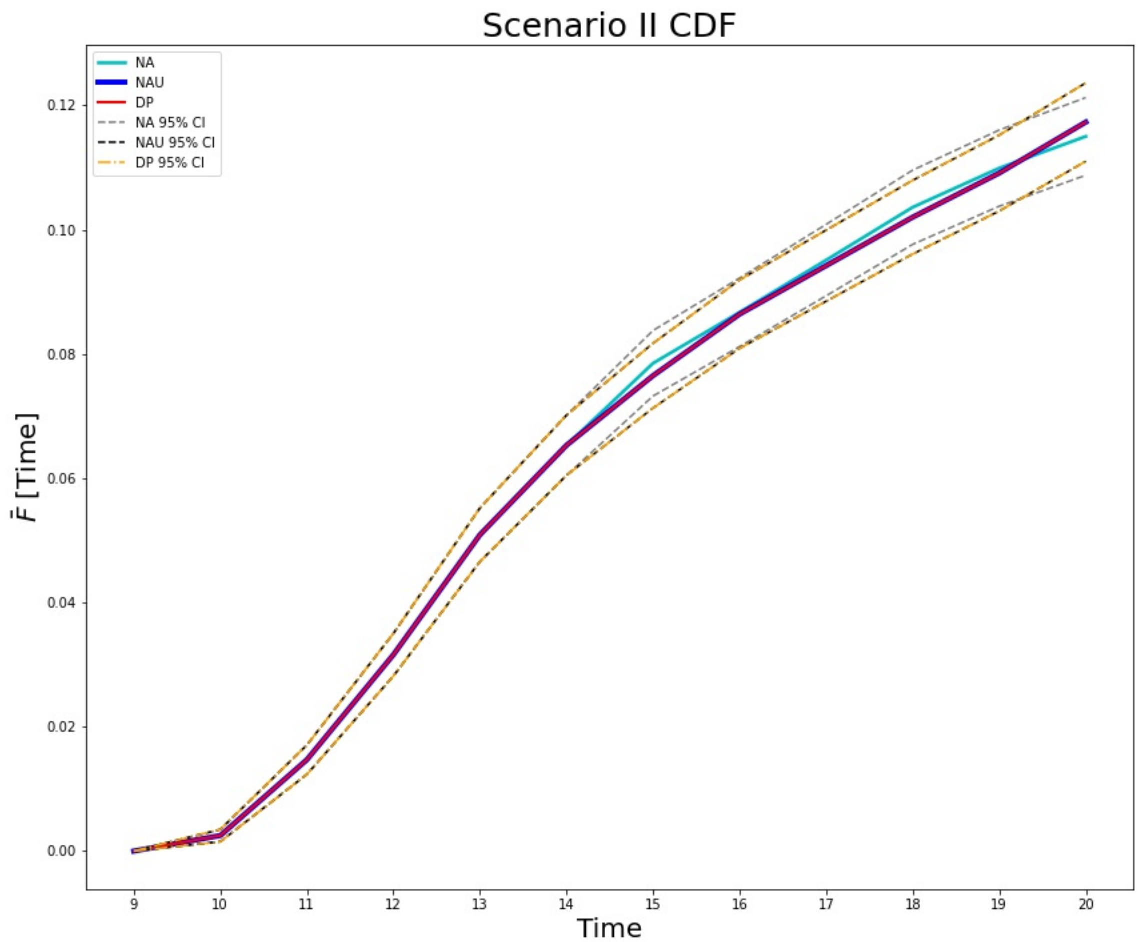

Figure 5: Empirical CDF for Scenario II. The CDF shows the empirical probability of finding Red in or before the time period on the x-axis. Since it takes 9 time periods for Blue and Red to be in the same area, the probability of finding Red before time period 9 is 0.

5.3 Scenario III

Next, we consider the case where Red is detected in the southern end of the SR and in one of the four middle cells distributed uniformly (see Figure 3, cells (20,9), (20,10), (20,11), (20,12)). We still assume the same mission set and resulting transition matrix as before as well as the same set of starting positions for Blue as the previous scenarios. Once again, each simulation replication simulates the starting positions for Red based on sampling from the Uniform distribution and Red's route using the probabilities in Red's transition matrix. The same search plans are used in all replications.

5.3.1 NA and NAU Results

Table 7 summarizes the search plans used for all the replications. In this scenario, the first time period where there is positive probability Red and Blue may be in the same cell is in time period 10; therefore, we present only the plans for time period 10 on.

Table 7: Scenario III NA and NAU Search Plans

Time Period	NA	NAU
10	(10,10)	(10,10)
11	(11,10)	(11,11)
12	(12,10)	(12,11)
13	(13,10)	(13,10)
14	(13,11)	(14,11)
15	(12,10)	(13,11)
16	(12,11)	(12,10)
17	(11,11)	(11,10)
18	(11,10)	(10,11)
19	(10,10)	(11,12)
20	(10,11)	(10,13)

The NA and NAU search plans yielded 1,051 and 1,124 detections, respectively, and corresponding 95% confidence intervals for the probability of detection of (.0991,0.1111) and (0.1062,0.1186), respectively.

5.3.2 DP Results

Table 8 presents the search pattern generated and used in all the replications. This search plan resulted in 1,190 detections and a 95% confidence interval for the probability of detection of (0.1127,0.1253).

Table 8: Scenario III DP Search Plan

Time Period	
10	(10,10)
11	(11,10)
12	(12,10)
13	(13,10)
14	(14,10)
15	(13,11)
16	(12,12)
17	(11,12)
18	(12,11)
19	(11,10)
20	(10,9)

5.3.3 Discussion

This scenario resulted in the largest improvement, so far, that the NAU algorithm gives over the NA algorithm. This scenario once again still results in the DP algorithm outperforming the NAU algorithm. Figure 6 presents the empirical CDF for this scenario.

We see a drop in the proportion of scenarios resulting in a detection. One may expect that since the starting position for Red lies far from cells in the SR from which Red may leave that we would see a higher proportion of Red detections, especially when compared to the first two scenarios. However, this is not what we observe. A possible explanation for this is that over half of the time available is spent transiting to areas where Red may be. Therefore, there are really only 10 periods where we may search, which is similar to the time available in scenario I. The empirical probability for Red leaving the SR during or before time period 10 is 25%.

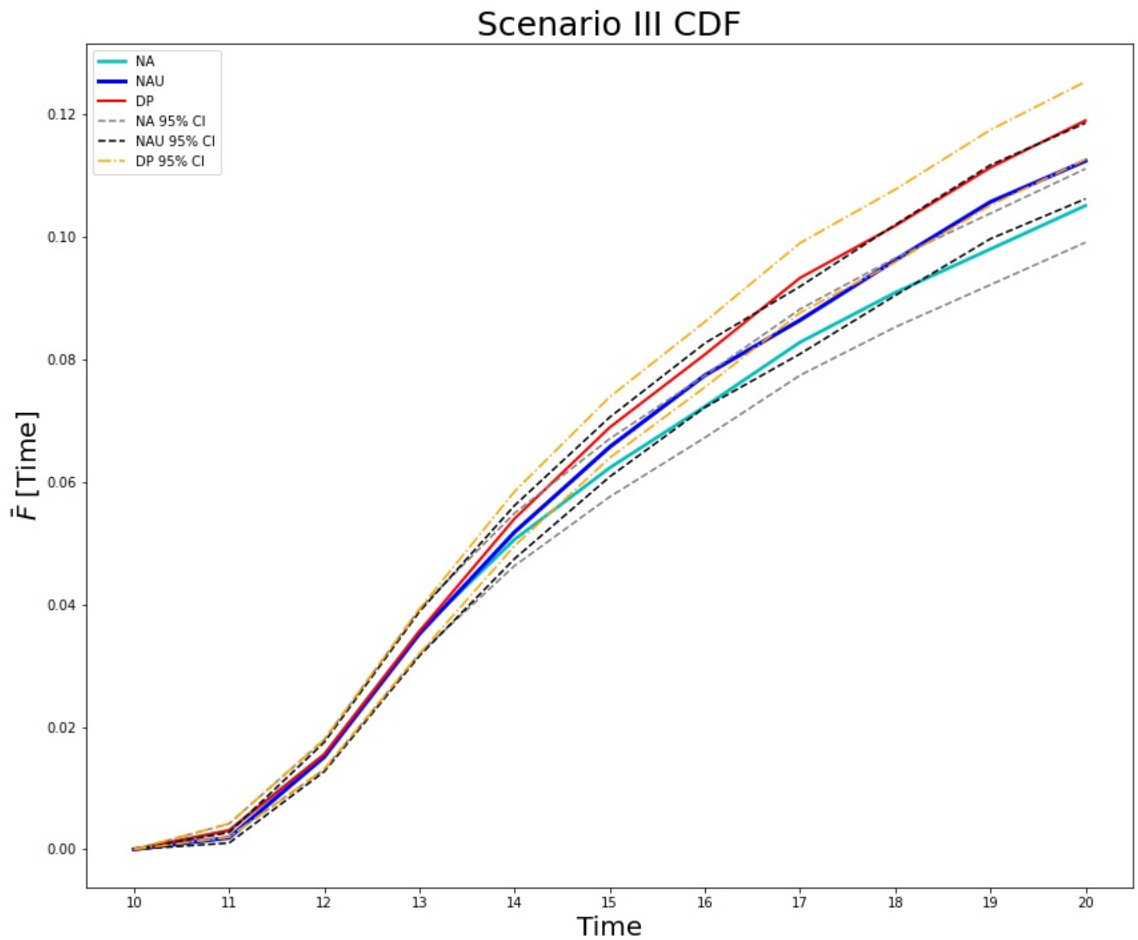

Figure 6: Empirical CDF for Scenario III. The CDF shows the empirical probability of finding Red in or before the time period on the x-axis. Since it takes 10 time periods for Blue and Red to be in the same area, the probability of finding Red before time period 10 is 0.

5.4 Scenario IV

Now we consider the case where Red is detected on the eastern end of the SR and in one of the four middle cells distributed uniformly (see Figure 3, cells (9,20), (10,20), (11,20), (12,20)). We maintain the same mission set and its associated transition matrix, and set of possible Blue starting positions as the first three scenarios. In each replication, the initial positions of Red and Red's route are simulated.

5.4.1 NA and NAU Results

Table 9 displays the search plans used in all the replications. Under this scenario, it is possible for Red and Blue to be in the same cell much sooner in time period 5 since Red starts more north than in previous scenarios.

Table 9: Scenario IV NA and NAU Search Plans

Time Period	NA	NAE
5	(5,19)	(5,19)
6	(6,19)	(6,19)
7	(7,19)	(7,19)
8	(6,19)	(8,19)
9	(6,18)	(7,19)
10	(6,19)	(6,18)
11	(5,19)	(5,18)
12	(5,18)	(4,18)
13	(4,18)	(3,18)
14	(3,18)	(2,18)
15	(2,18)	(1,18)
16	(1,18)	(1,17)
17	(1,19)	(2,17)
18	(1,18)	(1,18)
19	(1,17)	(1,17)
20	(1,18)	(1,16)

The NA and NAU search plans yielded 1,917 and 1,948 detections, respectively in the 10,000 replications, and corresponding 95% confidence intervals for the probability of detection of (0.1840,0.1994) and (0.1870,0.2026), respectively.

5.4.2 DP Results

Table 10 displays the search plan used in all the replications. This search plan resulted in 1,926 detections and a 95% confidence interval for the probability of detection of (0.1849,0.2003).

Table 10: Scenario IV DP Search Plan

Time Period	
5	(5,19)
6	(6,19)
7	(7,19)
8	(8,19)
9	(7,19)
10	(6,18)
11	(5,18)
12	(4,18)
13	(3,18)
14	(4,17)
15	(5,18)
16	(4,18)
17	(3,18)
18	(2,17)
19	(1,17)
20	(1,18)

5.4.3 Discussion

Once again we see improvements of the NAU algorithm over the NA. Additionally, this is the first scenario where we see the 95% confidence intervals for the probability of detection exceed 20%. The NAU algorithm slightly outperforms the DP algorithm for the first time.

It is interesting to see such a difference between these results and the results from scenario I. Under both, Red is starting in a cell at the edge of the SR with the same opportunities to leave the SR. However, in Scenario IV Red is starting north of its initial position in Scenario I. Thus it is possible for Blue to intercept and detect Red in less time resulting in less uncertainty concerning Red's location. Also, the empirical probability for Red leaving the SR during or before time period 5 is 44%, which is 13% lower than the 57% for Scenario I.

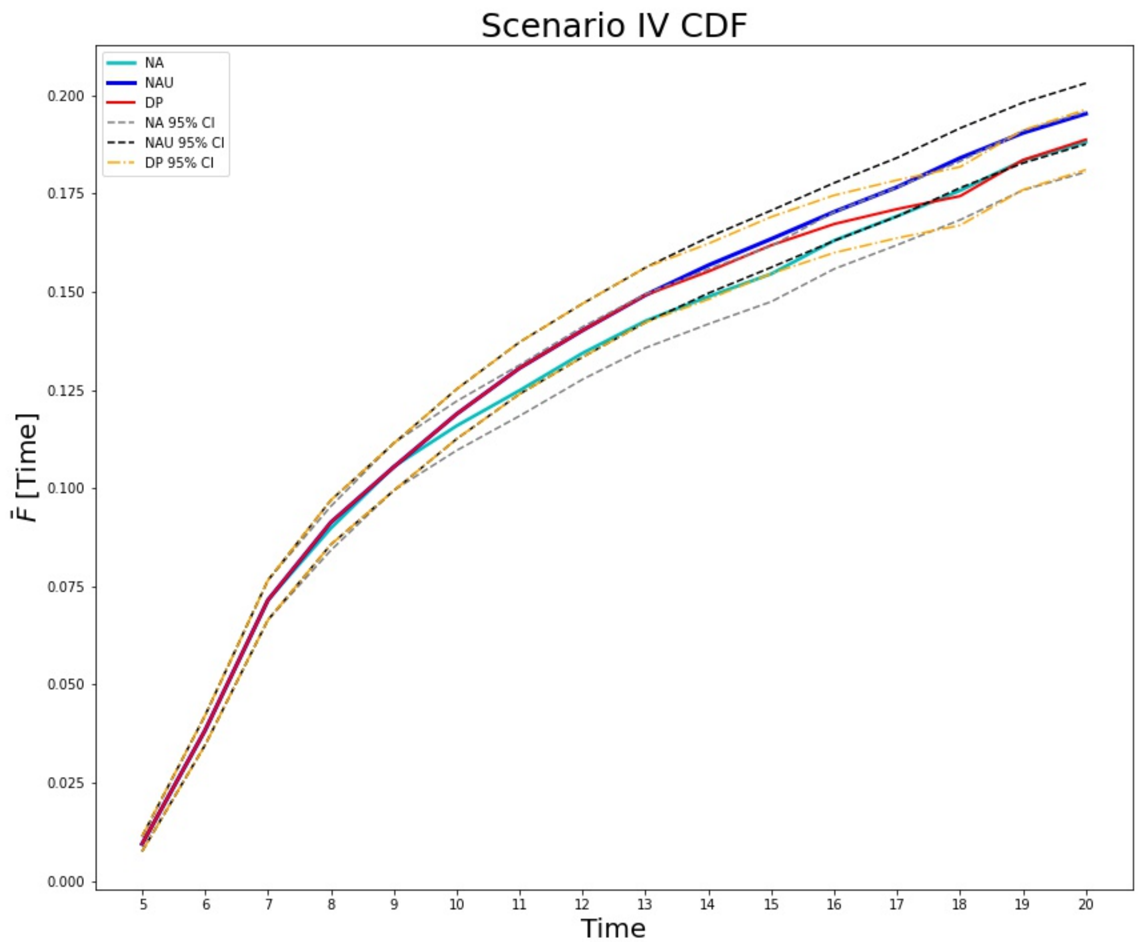

Figure 7: Empirical CDF for Scenario IV. The CDF shows the empirical probability of finding Red in or before the time period on the x-axis. Since it takes 5 time periods for Blue and Red to be in the same area, the probability of finding Red before time period 5 is 0.

5.5 Scenario V

Now we consider the case where Red is detected in a 4×4 cluster of cells in the middle of the SR (see Figure 3, cells (9,9), (9,10), (9,11), (9,12), (10,9), (10,10), (10,11), (10,12), (11,9), (11,10), (11,11), (11,12), (12,9), (12,10), (12,11), (12,12)). However, in this case we assume that Red has a different mission. For this scenario we assume that Red is conducting a patrol of the waterspace in the SR. Red may move in any direction, that is, it may move one cell to the northwest, north, northeast, west, east, southeast, south, or southwest, or remain in the same cell each time period, each with equal probability. Since we have intelligence Red is patrolling this waterspace, we assume that Red may not leave the SR during the 20 hour search time horizon. Table 11 displays a transition matrix for a similar scenario, except it is for a 3 by 3 SR as the case for a 20 by 20 SR results in a 400 $\times$ 400 transition matrix, which is difficult to interpret visually. The states in the transition matrix correspond to the (x,y) coordinate of the cells in the SR. Blue once again may start in any cell in the northernmost row of the SR. There are 10,000 replications, and each replication simulates

Table 11: Scenario V 9 Cell SR Transition Matrix

State	(1,1)	(1,2)	(1,3)	(2,1)	(2,2)	(2,3)	(3,1)	(3,2)	(3,3)
(1,1)	1/4	1/4	0	1/4	1/4	0	0	0	0
(1,2)	1/6	1/6	1/6	1/6	1/6	1/6	0	0	0
(1,3)	0	1/4	1/4	0	1/4	1/4	0	0	0
(2,1)	1/6	1/6	0	1/6	1/6	0	1/6	1/6	0
(2,2)	1/9	1/9	1/9	1/9	1/9	1/9	1/9	1/9	1/9
(2,3)	0	1/6	1/6	0	1/6	1/6	0	1/6	1/6
(3,1)	0	0	0	1/4	1/4	0	1/4	1/4	0
(3,2)	0	0	0	1/6	1/6	1/6	1/6	1/6	1/6
(3,3)	0	0	0	0	1/4	1/4	0	1/4	1/4

the initial position for Red and Red's route. The search plans are developed using the three algorithms. The same search plans are used in all replications.

5.5.1 Network NA and NAU Results

Table 12 displays the search plans used in all the replications. In this scenario, the first time period where there is positive probability Red and Blue may be in the same cell is in time period 4; therefore, we present only the results for time period 4 on.

Table 12: Scenario V NA and NAU Search Plans

Time Period	NA	NAE
4	(4,10)	(4,10)
5	(5,9)	(5,9)
6	(6,9)	(6,9)
7	(7,10)	(7,10)
8	(8,9)	(8,10)
9	(9,9)	(9,9)
10	(10,9)	(10,9)
11	(9,9)	(10,10)
12	(10,9)	(11,10)
13	(9,9)	(10,9)
14	(10,10)	(9,9)
15	(9,9)	(9,10)
16	(9,10)	(10,10)
17	(9,9)	(11,9)
18	(10,9)	(10,8)
19	(10,10)	(9,9)
20	(9,9)	(9,10)

The NA and NAU search plans yielded 1,647 and 1,694 detections, respectively, and corresponding 95% confidence intervals for the probability of detection of (0.1574,0.1720) and (0.1620,0.1768), respectively.

5.5.2 DP Results

Table 13 displays the generated search plan used for all the replications. This search plan resulted in 1,708 detections and a 95% confidence interval for the probability of detection of (0.1634,0.1782).

Table 13: Scenario V DP Search Plan

Time Period	
4	(4,9)
5	(5,9)
6	(6,10)
7	(7,10)
8	(8,9)
9	(9,9)
10	(10,10)
11	(10,9)
12	(11,9)
13	(10,10)
14	(9,10)
15	(9,9)
16	(10,9)
17	(11,10)
18	(10,11)
19	(9,11)
20	(8,10)

5.5.3 Discussion

The simulations in this scenario yield similar results to those of the other scenarios: the DP algorithm performs marginally better than the NAU algorithm, and both perform better than the NA algorithm.

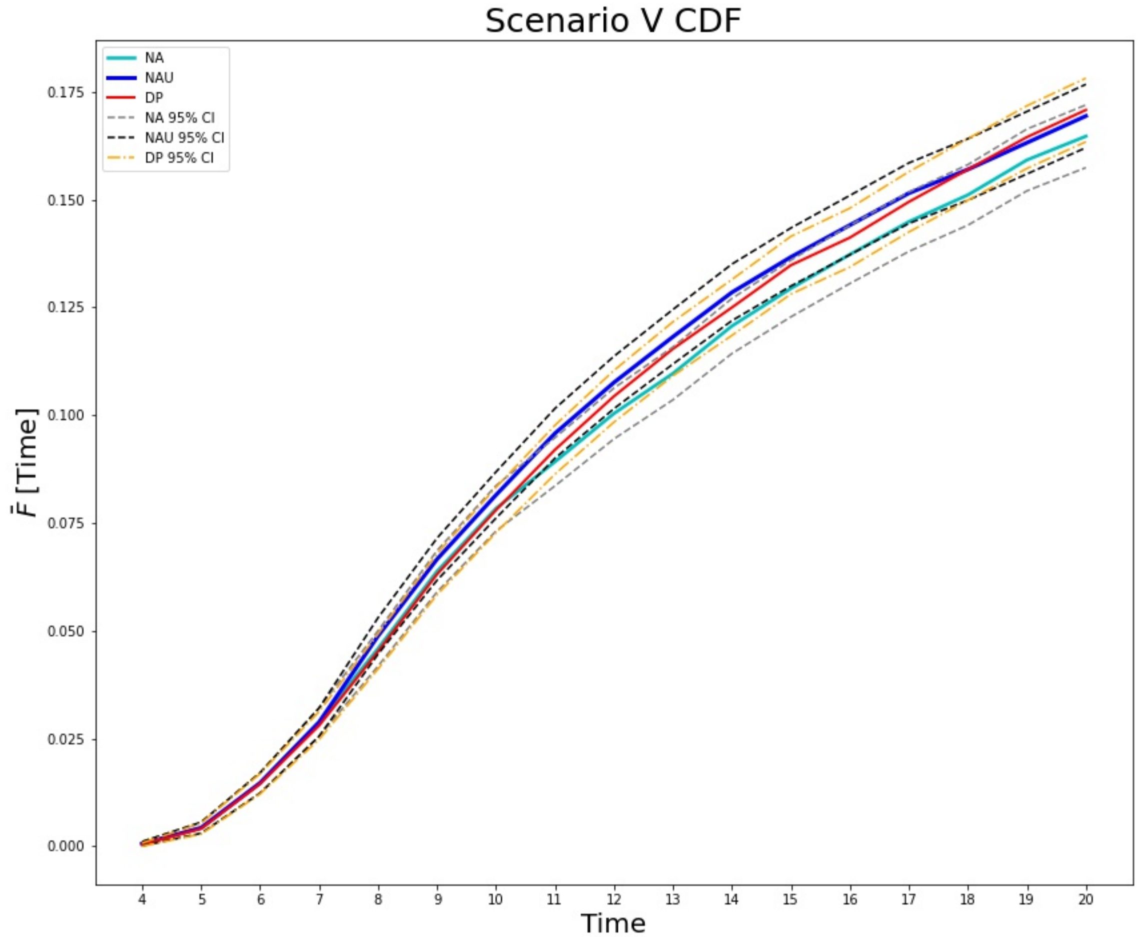

Figure 8: Empirical CDF for Scenario V. The CDF shows the empirical probability of finding Red in or before the time period on the x-axis. Since it takes 4 time periods for Blue and Red to be in the same area, the probability of finding Red before time period 4 is 0.

5.6 Scenario VI

In this section, we adjust the transition matrix to reflect how intelligence regarding Red's mission affects the probability of detection. For the scenario, we assume that Red starts in the most southeastern cell in the SR as in scenario I (see 3) and it may not leave the SR during the 20 hour search time period. In the scenario, we vary the probabilities in the transition matrix (M) to reflect different certainties in Red's movement. Specifically, for each cell, we make the probability of transitioning to the cell to the northwest to vary between 0.65, 0.85, and 0.95, and 0.98. The probability of transitioning to the north, northeast, east, west, and staying in the same cell are then set to be the remaining probability split evenly amongst the possible moves. Increasing the probability as specified reduces the variability in the routes that Red may take. Blue starts the search in the northernmost row of the SR. 10,000 replications simulate the initial position for Red and Red's route. The search plans are developed using the three algorithms. The same search plans are used in all replications.

The empirical CDF shown in Figure 9 leads to three observations. The first is that as

Blue becomes more certain of Red's movement, the probability of detecting Red in time period 10 improves, which intuitively makes sense and is not very interesting. Time period 10 is the first period where Blue is able to detect Red.

A more interesting observation from Figure 9 is that the NA algorithm does not detect Red for several time periods if Red is not found in time period 10 whereas the other two algorithms continue to find Red. This can be explained by the fact that the NA algorithm does not perform any updates. As the probability of transitioning to the northwest becomes higher and higher, the chances of Red coming back to the original northwesterly track becomes smaller because once it has deviated by at least one cell it is likely to continue northwest but on a different route. Put another way, if Red has deviated from the northwesterly route by 1 cell, it is more likely that it will follow a parallel path to the northwest and not return to the original route. Since the NA algorithm does not perform any updates for failing to find Red, it becomes difficult to find Red using this algorithm. However, with the NAU and DP algorithms, when Blue fails to find Red in time period 10, the algorithms infer that Red is no longer on the original route and perform updates to account for this which allow for further detections.

The third observation is the large gap between the performance of the NAU and DP algorithm in the 95% scenario. In the other scenarios, the search plans of the two algorithms follow each other closely throughout the search. However, in the 95% scenario there is a significant gap between the two.

Table 14 presents 95% confidence intervals for the probability of detection for each probability

Table 14: Scenario VI 95% Confidence Intervals for Probability of Detection for Various Probabilities of a Northwest Red Transition

Probability	NA	NAU	DP
65%	(0.1878, 0.2034)	(0.2654, 0.2828)	(0.2654, 0.2828)
85%	(0.2992, 0.3174)	(0.4780, 0.4976)	(0.4771, 0.4967)
95%	(0.7303, 0.7475)	(0.8286, 0.8431)	(0.6927, 0.7107)
98%	(0.8983, 0.9100)	(0.9456, 0.9542)	(0.9347, 0.9427)

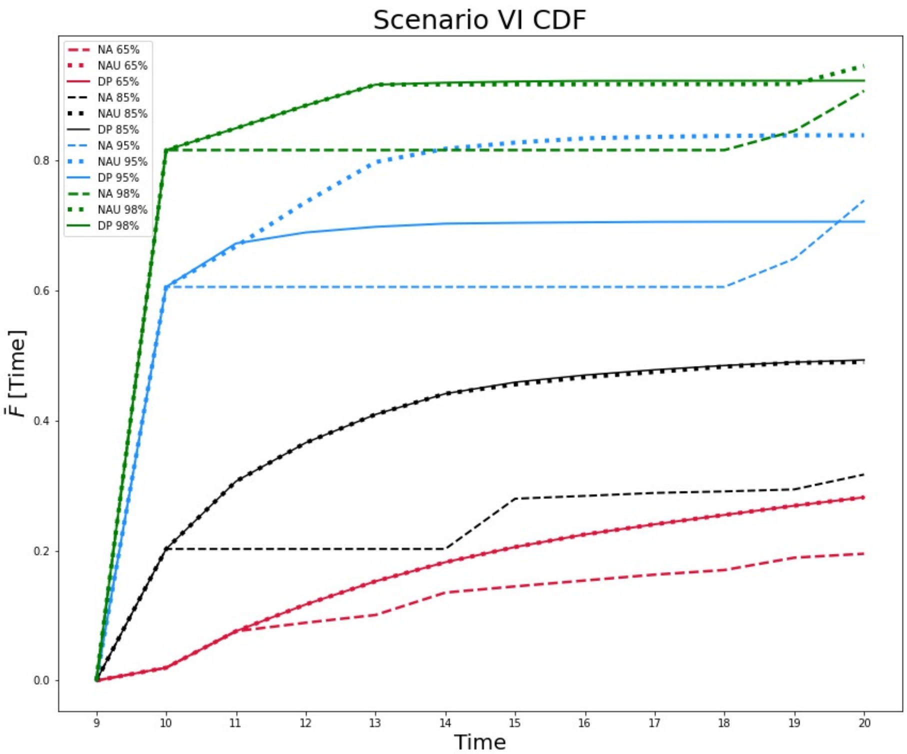

Figure 9: Empirical CDF for Scenario VI. The CDF shows the empirical probability of finding Red in or before the time period on the x-axis. Since it takes 10 time periods for Blue and Red to be in the same area, the probability of finding Red before time period 10 is 0. The CDF shows that as Blue becomes more certain of Red's movement, Blue has a higher probability of finding Red.

5.7 Computational Results

For each of the first five scenarios and for each algorithm we determined the computational cost for developing the search plan in CPU cycles. The results are summarized in Table 15. The DP algorithm was clearly the most efficient computationally of the three algorithms. Though this is true, we observe that as there are more time periods where updates are performed (due to different starting positions for Red) in scenarios IV and V, the DP algorithm nearly doubles in the number of cycles required. We see this trend as well in the NAU algorithm. However, the NA algorithm remains consistent throughout the scenarios, which is an expected result since the NA algorithm performs no updates.

Table 15: CPU Cycle Comparison

Scenario	NA	NAU	DP
I	3.98×10^{11}	1.53×10^{12}	2.72×10^{9}
II	3.89×10^{11}	1.75×10^{12}	2.80×10^{9}
III	4.00×10^{11}	1.51×10^{12}	2.46×10^{9}
IV	3.88×10^{11}	2.91×10^{12}	4.40×10^{9}
V	4.09×10^{11}	3.27×10^{12}	4.88×10^{9}

Due to the smaller computational cost of the DP algorithm, it may be possible that the two time period horizon may be increased to include more time periods before reaching the computational costs of the other algorithms, and this may lead to improved performance of the algorithm. We leave this for future work.

6 Conclusions and Future Research

Due to advancements in platforms and increased activity by strategic adversaries, the United States' dominance in the underwater domain is at risk. The ability to quickly locate adversarial submarines provides flexibility in the deployment of the USN strategic submarines, and it also provides security from attacks against the homeland.

In this report we address the need for a refined search tool available to decision makers aboard submarines to aid in their search for adversarial submarines. We study different approaches for a basic search algorithm that is capable of being further developed and implemented aboard submarines.

Our research begins with modeling Red submarine movement as a Discrete Time Markov Chain, where the states correspond to cells in the SR and the transition matrix corresponds to the probability of moving from cell i to cell j. Red's transition matrix defines the mission Red is conducting. We assume for our research that both Red's starting position probability distribution and its transition matrix are known by Blue.

We then define three algorithms to determine an optimal search route for Blue given probabilistically Red's starting position and transition matrix as well as the starting area for Blue at one edge of the SR. The first algorithm, the NA, maximizes the probability of detecting Red over the time horizon using a network optimization framework. The search plan is calculated once and without updates, assuming that Red is not detected in a searched cell. We then expand upon this algorithm by performing a Bayesian update of the position of Red after each cell search assuming that the search is unsuccessful and call this algorithm the NAU; following the Bayesian update a new search plan that maximizes the probability of detecting Red in the remaining search period is computed. The third algorithm, the DP, also makes use of Bayesian updates which assume the cell search is unsuccessful; however, in this algorithm instead of maximizing the probability of detecting Red over the entire remaining planning horizon, we maximize the probability of finding Red in the next time period or failing that in the time period after that. Besides the difference in how many time periods for which we are looking ahead, another fundamental difference between the NAU and DP algorithms is when the Bayesian updates are performed. In the NAU algorithm, we perform the update once each time period to update the probability distribution for Red's location to account for Red not being in the cell that was directed to be searched in the current time period; the update is used to determine the entire future search path. In the DP algorithm there are two updates; once to determine which cell will maximize finding Red in two time steps assuming Red was not found in the next time step and then a second time in a manner similar to the NAU algorithm. Both the NAU and DP search plans are computed before search begins.

To compare the three algorithms, we pose five scenarios where we vary the starting location for Red and the number of cells in which it may initially be located. The transition matrix governing Red's movement is the same in the first four scenarios; it permits Red to either remain in the same cell or move to adjacent cells that are to the north, northeast, northwest, east, or west of Red's current cell. The fifth scenario's transition matrix allows for Red to travel to any adjacent cell or remain in the same cell. In a sixth scenario, we keep Red's starting position fixed in one cell, but we vary the probability of transitioning between cells as a way to represent the effect better intelligence regarding Red's mission has on the probability of detecting Red. For each scenario, 10,000 initial starting positions (in cases where it is possible for Red to start in more than one cell) and Red routes are simulated. The same NA, NAU, and DP search plans are used in all replications for a scenario. Using the optimal search plans and the simulated routes for Red, we determine if Blue detects Red and during which period the search plan detects Red. We compute 95% confidence intervals for the probability of detecting Red and generate empirical CDFs to illustrate how the probability of detecting Red varies with time to compare the three algorithms. Additionally, for the first five scenarios we calculate the computational time required to generate the search plan for each algorithm in CPU cycles.

The results of the first five scenarios are consistent. The NAU and DP algorithms outperform the NA algorithm. This observation is reasonable because the NAU and DP search plans incorporate the pessimistic information that Red is never detected. The DP algorithm outperforms the NAU algorithm in four of the first five scenarios; however, in all five scenarios the 95% confidence intervals for the probability of detection overlap, meaning their performances are comparable. The computational results show that the DP algorithm requires the least computational power to generate a search plan.

In the sixth scenario results show that as Blue becomes more certain of Red's intended course, the probability Blue finds Red in the first time period where it is possible for both submarines to be in the same cell of the SR increases. However, Figure 9 shows that the more certain we are of Red's path, the harder it becomes to find Red if Red does not adhere to the most probable path.

Because the performance of the NAU and DP algorithms are so close, we turn to the computational cost of determining the search plan for the method we recommend as the algorithm to be further explored and developed. Since the DP algorithm requires much less computational cost and in most scenarios outperforms the NAU algorithm, we recommend the DP algorithm as the best algorithm we consider in our research.

We recommend in future research to explore non-perfect sensors where there are both false positive and false negative search results, and where the probability of detecting Red if both submarines in the same cell is less than one. This will allow for more realistic conditions where environmental factors play into how well an adversarial submarine may be detected.

We also leave for future work relaxing the assumptions that the transition matrix is known by Blue and using concepts from Game Theory to explore worst-case transition matrices for Blue.

Finally, applying the work in this report to multiple search assets conducting the search concurrently is an important problem that may well require a different modeling approach.

References

[Benkoski et al., 1991] Benkoski, S., Monticino, M., and Weisinger, J. (1991). A survey of the search theory literature. *Naval Research Logistics*, 38(4):469–494.

[Brown, 1980] Brown, S. (1980). Optimal search for a moving target in discrete time and space. *Operations Research*, 28(6):1275–1289.

[Cares, 2021] Cares, J. (2021). The fundamentals of Antisubmarine Warfare (ASW) doctrine. Technical Report SORG Memorandum 21-04, Submarine Operations Research Group.

[Charnes and Cooper, 1958] Charnes, A. and Cooper, W. (1958). The theory of search: Optimum distribution of search effort. *Management Science*, 5(1):44–50.

[Dell et al., 1996] Dell, R., Eagle, J., Martins, G., and Santos, A. (1996). Using multiple searchers in constrained-path, moving target search problems. *Naval Research Logistics*, 43(4):463–480.

[Dobbie, 1974] Dobbie, J. (1974). A two cell model of search for a moving target. *Operations Research*, 22(1):79–92.

[Foraker, 2011] Foraker, J. (2011). *Optimal Search For Moving Targets in Continuous Time and Space Using Consistent Approximations*. Doctoral dissertation, Department of Operations Research, Naval Postgraduate School, Monterey, CA.

[Iida, 1989] Iida, K. (1989). Optimal search plan minimizing the expected risk of the search for a target with conditionally deterministic motion. *Naval Research Logistics*, 36(5):597–613.

[Kan, 1977] Kan, Y. (1977). Optimal search of a moving target. *Operations Research*, 25(5):864–870.

[Koopman, 1946] Koopman, B. (1946). Search and screening. Technical Report OEG Report No. 56, Operations Evaluation Group Office of the Chief of Naval Operations.

[Koopman, 1956a] Koopman, B. (1956a). The theory of search, part i. kinematic bases. *Operations Research*, 4(3):324–346.

[Koopman, 1956b] Koopman, B. (1956b). The theory of search, part ii. target detection. *Operations Research*, 4(5):503–531.

[Koopman, 1957] Koopman, B. (1957). The theory of search, part iii. the optimum distribution of searching effort. *Operations Research*, 5(5):613–626.

[Koopman, 1980] Koopman, B. (1980). *Search and Screening: General Principles with Historical Applications*. Pergamon, New York.

[LaGrone, 2018] LaGrone, S. (2018). Navy reestablishes U.S. 2nd Fleet to face Russian threat; plan calls for 250 person command in Norfolk. *USNI News*. https://news.usni.org/2018/05/04/navy-reestablishes-2nd-fleet-plan-calls-for-250-person-command-in-norfolk.

[Nakai, 1973] Nakai, T. (1973). A model of search for a target moving among three boxes: Some special cases. *Journal of the Operations Research Society of Japan*, 16:151–162.

[Pollock, 1970] Pollock, S. (1970). A simple model of search for a moving target. *Operations Research*, 18(5):883–903.

[Richardson and Stone, 1971] Richardson, H. and Stone, L. (1971). Operations analysis during the underwater search for scorpion. *Naval Research Logistics Quarterly*, 18(2):141–157.

[Royset and Sato, 2010] Royset, J. and Sato, H. (2010). Route optimization for multiple searchers. *Naval Research Logistics*, 57(8):701–717.

[Sato and Royset, 2010] Sato, H. and Royset, J. (2010). Path optimization for the resource-constrained searcher. *Naval Research Logistics*, 57(5):422–440.

[Schweitzer, 1971] Schweitzer, P. (1971). Threshold probabilities when searching for a moving target. *Operations Research*, 19(3):707–709.

[Shelbourne, 2021] Shelbourne, M. (2021). Navy creates new Atlantic Destroyer Task Group to hunt Russian submarines. *USNI News*. https://news.usni.org/2021/09/27/navy-creates-new-atlantic-destroyer-task-group-to-hunt-russian-submarines.

[Stone, 1977] Stone, L. (1977). Search for targets with generalized conditionally deterministic motion. *SIAM Journal of Applied Mathematics*, 33(3):456–468.

[Stone et al., 2014] Stone, L., Keller, C., Kratzke, T., and Strumpfer, J. (2014). Search for the wreckage of air france flight af 447. *Statistical Science*, 29(1):69–80.

[Stone and Richardson, 1974] Stone, L. and Richardson, H. (1974). Search for targets with conditionally deterministic motion. *SIAM Journal of Applied Mathematics*, 27(2):239–255.

[Washburn, 1998] Washburn, A. (1998). Branch and bound methods for a search problem. *Naval Research Logistics*, 45(3):243–257.