



Calhoun: The NPS Institutional Archive
DSpace Repository

Theses and Dissertations

1. Thesis and Dissertation Collection, all items

2018-09

**MARITIME DOMAIN AWARENESS BY ANOMALY
DETECTION LEVERAGING TRACK INFORMATION**

Ashraf, Tauseef

Monterey, CA; Naval Postgraduate School

<http://hdl.handle.net/10945/60384>

Downloaded from NPS Archive: Calhoun



Calhoun is a project of the Dudley Knox Library at NPS, furthering the precepts and goals of open government and government transparency. All information contained herein has been approved for release by the NPS Public Affairs Officer.

Dudley Knox Library / Naval Postgraduate School
411 Dyer Road / 1 University Circle
Monterey, California USA 93943

<http://www.nps.edu/library>



**NAVAL
POSTGRADUATE
SCHOOL**

MONTEREY, CALIFORNIA

THESIS

**MARITIME DOMAIN AWARENESS BY ANOMALY
DETECTION LEVERAGING TRACK INFORMATION**

by

Tauseef Ashraf

September 2018

Thesis Advisor:

Frank E. Kragh

Co-Advisor:

James W. Scrofani

Approved for public release. Distribution is unlimited.

THIS PAGE INTENTIONALLY LEFT BLANK

REPORT DOCUMENTATION PAGE			<i>Form Approved OMB No. 0704-0188</i>	
Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington, DC 20503.				
1. AGENCY USE ONLY (Leave blank)		2. REPORT DATE September 2018	3. REPORT TYPE AND DATES COVERED Master's thesis	
4. TITLE AND SUBTITLE MARITIME DOMAIN AWARENESS BY ANOMALY DETECTION LEVERAGING TRACK INFORMATION			5. FUNDING NUMBERS	
6. AUTHOR(S) Tauseef Ashraf				
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943-5000			8. PERFORMING ORGANIZATION REPORT NUMBER	
9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES) N/A			10. SPONSORING / MONITORING AGENCY REPORT NUMBER	
11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.				
12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release. Distribution is unlimited.			12b. DISTRIBUTION CODE A	
13. ABSTRACT (maximum 200 words) <p>Techniques for anomaly detection in the maritime domain are developed in this thesis using an area metric that measures the degree of similarity, or distinction, between ships' tracks using the area between ships' tracks. A modified k-means algorithm is applied to the areas calculated between ships tracks to extract sea lanes and classify these ships' tracks into sea lanes. Bayes' theorem of conditional probabilities is used to calculate the probability of ship tracks being in different sea lanes. Two types of anomaly detection are examined in this thesis. Ships not sailing in standard sea lanes are detected using Bayes' theorem, and ships following one another are detected using user-defined thresholds on the minimum area between sea tracks. The development of the metric that compares ships' track data using area is the most significant result of this thesis.</p>				
14. SUBJECT TERMS area calculation algorithm, Bayes' theorem, area metric, k-means cluster, anomaly detection, sea lane, ship track, conditional probability			15. NUMBER OF PAGES 125	
			16. PRICE CODE	
17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT UU	

THIS PAGE INTENTIONALLY LEFT BLANK

Approved for public release. Distribution is unlimited.

**MARITIME DOMAIN AWARENESS BY ANOMALY DETECTION
LEVERAGING TRACK INFORMATION**

Tauseef Ashraf
Lieutenant Commander, Pakistan Navy
BEE, National University of Science and Technology (Pakistan), 2011

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN ELECTRICAL ENGINEERING

from the

**NAVAL POSTGRADUATE SCHOOL
September 2018**

Approved by: Frank E. Kragh
Advisor

James W. Scrofani
Co-Advisor

R. Clark Robertson
Chair, Department of Electrical and Computer Engineering

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

Techniques for anomaly detection in the maritime domain are developed in this thesis using an area metric that measures the degree of similarity, or distinction, between ships' tracks using the area between ships' tracks. A modified k-means algorithm is applied to the areas calculated between ships tracks to extract sea lanes and classify these ships' tracks into sea lanes. Bayes' theorem of conditional probabilities is used to calculate the probability of ship tracks being in different sea lanes. Two types of anomaly detection are examined in this thesis. Ships not sailing in standard sea lanes are detected using Bayes' theorem, and ships following one another are detected using user-defined thresholds on the minimum area between sea tracks. The development of the metric that compares ships' track data using area is the most significant result of this thesis.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

I.	INTRODUCTION.....	1
A.	RESEARCH OBJECTIVE	2
B.	LITERATURE REVIEW	2
C.	ORGANIZATION OF THESIS	3
II.	BACKGROUND	5
A.	CONCEPT OF MARITIME DOMAIN AWARENESS	5
B.	ANOMALY DETECTION	6
C.	AIS DATA.....	8
D.	PREPROCESSING OF DATA.....	9
E.	K-MEANS CLUSTERING.....	9
1.	Perform Cluster Assignment	11
2.	Update the Centroid	11
III.	ALGORITHMS AND METHODS FOR CLASSIFICATION AND ANOMALY DETECTION	17
A.	AREA METRIC.....	18
B.	DEFINING A SEA LANE USING AREA METRIC AND MODIFIED K-MEANS CLUSTERING ALGORITHM.....	24
1.	Determining Number of Sea Lanes to Be Defined	24
2.	Finding Areas between Ships' Tracks.....	24
3.	Clustering the Ships' Tracks Using Modified <i>k</i>-means Algorithm.....	26
C.	CLASSIFICATION OF SHIPS' TRACKS BY SEA LANE.....	28
1.	User-Defined Threshold	28
2.	The Modified <i>k</i>-means Clustering Algorithm	29
3.	Bayes' Theorem.....	30
D.	ANOMALY DETECTION USING AREA METRIC.....	35
1.	Detecting Outliers in Standard Sea Lanes.....	35
2.	Detecting Group of Ships Following or Parallel in a Sea Lane.....	35
IV.	RESULTS	39
A.	GENERATING SHIP'S TRACK DATA.....	39
B.	AREA METRIC ALGORITHM BETWEEN SHIPS' TRACKS AND SEA LANE.....	40
C.	DEFINING A SEA LANE USING AREA METRIC AND MODIFIED K-MEANS ALGORITHM	41

D.	CLASSIFICATION OF SHIPS' TRACKS INTO SEA LANES USING AREA METRIC ALGORITHM	48
1.	User-Defined Threshold	49
2.	The Modified k -means Clustering Algorithm	50
3.	Bayes' Theorem.....	52
E.	ANOMALY DETECTION	60
1.	Detecting Outliers in Standard Sea Lanes Using Area Metric and Bayes' Theorem.....	60
2.	Detecting Ships Following One Another Using Area Metric Algorithm and User-Defined Threshold.....	61
V.	CONCLUSION	65
A.	SIGNIFICANT CONTRIBUTIONS.....	65
B.	FUTURE WORK.....	66
	APPENDIX.....	69
	LIST OF REFERENCES.....	103
	INITIAL DISTRIBUTION LIST	107

LIST OF FIGURES

Figure 1.	A Typical AIS Display. Source: [3].....	1
Figure 2.	Types of Anomalies. Source: [16].	7
Figure 3.	k -means Clustering Algorithm Flowchart.....	10
Figure 4.	Data Points of Individual Scores.....	12
Figure 5.	Data Points with Initially Selected Centroids	13
Figure 6.	Final Clusters Using k -means Algorithm.....	15
Figure 7.	Positional Reports of Two Ships on Same Path.....	17
Figure 8.	Area between Sea Lane and Ship’s Track	19
Figure 9.	A Triangle Represented by Three Points	20
Figure 10.	Euclidian Distance d between Points P and Q	20
Figure 11.	Area between Ship’s Track and Sea Lane	21
Figure 12.	Algorithm for Calculating Area Metric	22
Figure 13.	Crossover of a Sea Lane and a Ship’s Track	23
Figure 14.	Flowchart for Defining Sea Lane from Ships’ Tracks.....	25
Figure 15.	Flowchart for Classifying Ships’ Tracks into Sea Lanes.....	29
Figure 16.	Venn Diagram for Bayes’ Probabilities Theorem	31
Figure 17.	PDF for Ships’ Tracks $f_{X,Y}(x,y)$ —Side View	34
Figure 18.	PDF for Ships’ Tracks $f_{X,Y}(x,y)$ —Top View	34
Figure 19.	Flowchart for Anomaly Detection—Ships Following One Another in a Sea Lane	37
Figure 20.	Sea Lane between Points A and B	39
Figure 21.	Ships’ Tracks and a Sea Lane from Point A to B	40

Figure 22.	Ships' Tracks in Sea Lanes A and B	42
Figure 23.	Ships' Tracks Classified into Sea Lane A and B	45
Figure 24.	Sea Lane A and Sea Lane B formed by Averaging Tracks Belonging to these Sea Lanes	48
Figure 25.	Sea Lanes A and B with Ships' Tracks	49
Figure 26.	Classifying Ships' Tracks into Sea Lanes Using Modified k -means Algorithm	51
Figure 27.	Ships' Tracks and Two Sea Lanes	53
Figure 28.	Training Data of Ships' Tracks in Sea lanes A or B	56
Figure 29.	Test Data Set of Ships' Tracks and Sea Lanes A and B	58
Figure 30.	Ships' Tracks in the Same Sea Lane in Spatial Domain.....	62
Figure 31.	Ships' Tracks in the Same Sea Lane at Different Times	62
Figure 32.	Anomaly Detection of Ships Following One Another.....	63

LIST OF TABLES

Table 1.	Data Points to be Grouped Using k -means Algorithm.....	12
Table 2.	Centroids of Data Points	13
Table 3.	Data Points with New Centroids.....	14
Table 4.	Euclidian Distance of Each Data Point from Centroid.....	14
Table 5.	Final Clusters Using k -means Algorithm.....	15
Table 6.	Example of Areas between Ships' Tracks (in NM^2).....	26
Table 7.	Cluster Assignment Based on Minimum Area Metric.....	27
Table 8.	Area of Ships' Tracks with Respect to Sea Lane.....	41
Table 9.	Area Metric between Ships' Tracks (in NM^2)	43
Table 10.	Sea Lane Assignment Based on Minimum Area Metric	44
Table 11.	Classification of Ships' Tracks into Sea Lanes Based on Minimum Value of S (Sum of Smallest Area Metrics)	46
Table 12.	Classification of Ships' Tracks into Sea Lanes Using a User-Defined Threshold	50
Table 13.	Ships' Classification into Sea Lanes Using Modified k -means Algorithm.....	52
Table 14.	Area of Ships' Tracks with Respect to Sea Lanes A and B	53
Table 15.	Parameters of Conditional PDFs.....	54
Table 16.	Probabilities of Sea Lane A and Sea Lane B Given Ships' Track Areas	55
Table 17.	Parameters of Conditional PDFs Computed from Training Data.....	57
Table 18.	Posterior Probabilities of Sea Lane A and Sea Lane B Given Ships' Tracks Areas	59
Table 19.	Anomaly Detection Based on Posterior Probabilities of Sea Lane A and Sea Lane B Given Ships' Tracks Areas	61

Table 20. Area Metric between Ships' Tracks (in NM^2).....64

LIST OF ACRONYMS AND ABBREVIATIONS

AIS	Automatic Identification System
GMM	Gaussian Mixture Model
IMO	International Maritime Organization
ITU	International Telecommunication Union
KDE	Kernel Density Estimator
KF	Kalman Filter
MDA	Maritime Domain Awareness
PDF	Probability Density Function
RMP	Recognized Maritime Picture
VHF	Very High Frequency
NM	Nautical Miles

THIS PAGE INTENTIONALLY LEFT BLANK

ACKNOWLEDGMENTS

I am indebted to Professor Frank Kragh for his advice and support during every stage of this thesis; without his valued input, this thesis may not have existed. Professor Frank Kragh's statistical insight was key to the successful completion of the thesis. I would also like to extend my gratitude to Professor Jim Scrofani for his guidance and contribution to this research work.

I would like to thank NPS and the Electrical and Computer Engineering faculty for the great learning experience I have been provided and the U.S. IMET program for funding my study. I am also indebted to my country, Pakistan, and the Pakistan Navy for providing me the educational opportunity in the USA.

I would like to extend a very special thanks to my wife, Mona, for her support, kindness, love and sacrifice. She spared me time to work on my thesis by taking care of our home and three children. Her consistent support and encouragement enabled me to complete my research work.

I would like to dedicate my work to my parents, who have always encouraged me to work hard and pursue my dreams. Words cannot express my gratitude for all the sacrifices my parents have made to raise and educate me.

THIS PAGE INTENTIONALLY LEFT BLANK

I. INTRODUCTION

In the modern world, most trade is done through maritime shipping, and its security is considered paramount for continued and uninterrupted sea lane transportation. More than 80 percent of global trade is transported through maritime shipping [1]. Territorial waters are mostly guarded by regional security forces including coast guards and respective navies, but in open seas and oceans, maritime traffic is mostly unguarded and vulnerable; thus, the need for an automated monitoring system of global maritime traffic to improve maritime security is essential.

The International Maritime Organization (IMO), a specialized agency of the United Nations, has been mandated to regulate shipping [1]. In 2002, the IMO decided that all ships over 300 tons and all passenger vessels must be fitted with Automatic Identification System (AIS) transceivers to improve security in the maritime environment [2]. AIS allows all vessels in territorial waters to see other ships' identifiers and kinematic data including their names, types, identifiers, destinations, locations, headings, speed, etc. Different hardware and software are available onboard ships to graphically represent the maritime picture using received AIS data, as illustrated in Figure 1.

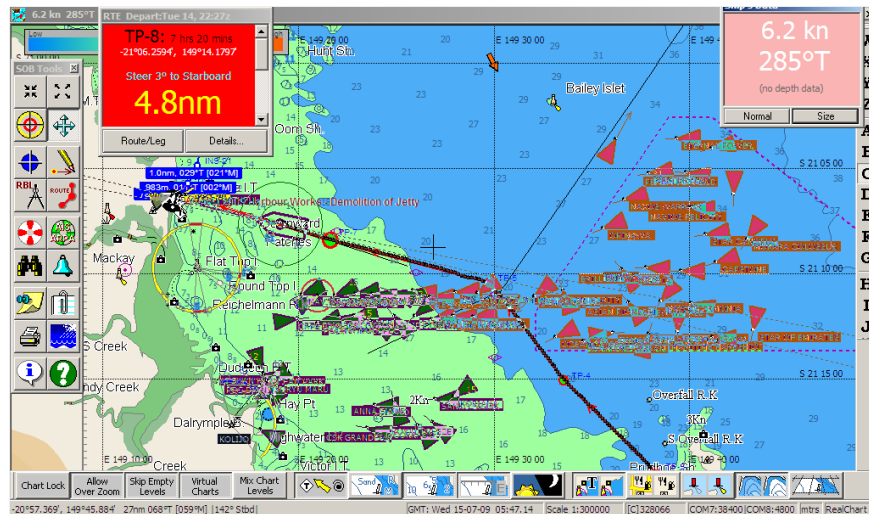


Figure 1. A Typical AIS Display. Source: [3].

The same AIS data is used to track and predict ships' future positions by maritime security organizations. Due to the huge volume of maritime traffic, manual monitoring for detecting anomalies is very challenging. Development of an automatic monitoring and anomaly detection system for the maritime domain will not only help to efficiently allocate limited human and material resources but will also help to improve the security of life and ships at sea. Early detection of anomalies, if determined to be nefarious, may also foster timely reactions and avoid mishaps or accidents.

A. RESEARCH OBJECTIVE

The objective of this work is to explore methods of defining sea lanes using position data of maritime traffic similar to AIS, sorting and classifying maritime traffic into different sea lanes using algorithms that are not sensitive to the frequency of position data points. In need of a metric or distance function to measure the degree of similarity or distinction between two ships' tracks, we chose the geographical area defined by the two tracks with a simple method of closing the area on the ends. This metric is fundamental to this work. Using this metric, we classify ships into sea lanes, detect ships exhibiting anomalous behavior when not following standard sea lanes, and detect ships following one another in a specific sea lane.

B. LITERATURE REVIEW

Much work has already been done in the field of Maritime Domain Awareness (MDA) in general and anomaly detection in MDA in particular. This literature review sheds light onto some of the related work. We have tried to fill in the gaps of these works by adapting a slightly different approach of problem solving.

Bayesian statistics are used in [4] to indicate five different anomalous ship behaviors. The overall probability of exhibiting anomalous behavior is computed through individual probabilities of each type of anomaly using a Bayesian networks.

Information regarding normal patterns of maritime traffic is extracted using an adaptive mixture based neural network, which is then used for outlier detection in [5]. Training data of maritime traffic is used to estimate the parameters of the multidimensional

probability density function (PDF), which is representative of a subnetwork in the neural network.

The kernel density estimator (KDE) and Gaussian mixture model (GMM) are used for anomaly detection in the maritime domain, and their individual performances are compared in [6]. Historical AIS data is used to train both the models and estimate the PDF of each sea lane, which are used for outlier detection.

In [7], positional report data of maritime traffic from various sensors (AIS, GPS, radar, etc.) are preprocessed for synchronization and classified into ships' tracks using a fuzzy k -nearest neighbor algorithm. Future position of ships' tracks is predicted through a Kalman Filter, where parameters of the filter are estimated using the Echo State Network (ESN).

The Hough Transformation is used with AIS data in [8] to extract a normal linear traffic pattern to generate normal sea lanes. The sea lanes are then split into sections, and a normal distribution is fit to these sections to find anomalous behavior of the vessels.

In [9], maritime traffic in a particular area is spatially clustered using k -means clustering. These clusters are then temporally analyzed to determine the kinematics of clusters as well as the change of clusters with time.

C. ORGANIZATION OF THESIS

A background discussion of the concept of MDA and its need in our present world, different types of anomalies existing in the maritime domain, and k -means clustering and data preprocessing are discussed in Chapter II. The metric using the geographical area between sea lanes and ships' tracks, using this metric to define sea lanes, classifying ships into sea lanes, and detecting anomalous behavior of ships, are discussed in Chapter III. Results and analysis of these algorithms are discussed in Chapter IV. A summary of the work performed and ideas for future work are provided in Chapter V. All the MATLAB code for implementing the algorithms discussed in this thesis are included in the Appendix.

THIS PAGE INTENTIONALLY LEFT BLANK

II. BACKGROUND

The concept of MDA originates from the fact that maritime security is attained through development of effective understanding of the events that are happening in the maritime domain [10], [11]. Development of a recognized maritime picture (RMP) by fusion of maritime data from various regional and international organizations and extracting actionable information from this RMP is the fundamental concept of MDA [12]. In a speech delivered in January 2002, President George W. Bush said, “The heart of the Maritime Domain Awareness program is accurate information, intelligence, surveillance, and reconnaissance of all vessels, cargo, and people extending well beyond our traditional maritime boundaries” [13]. Previously, the Department of Defense’s (DoD) focus in regard to MDA was primarily towards opposing navies; however, recently this focus has shifted more towards commercial and maritime threats due to commercial trade’s huge dependency on sea routes, increased piracy, and proliferation of weapons, all of which are paramount to national security [12], [14].

To gain a better understanding of the movement of vessels and to be able to automatically classify their movements as normal or abnormal behavior is a challenge. From a national security viewpoint, it is of particular interest to classify the outliers having suspicious or abnormal behavior out at sea. Using AIS installed on ships, we can now analyze the movements of vessels at sea; however, considering the amount of traffic at sea, automated analysis of vessel behavior seems to be the only way forward.

A. CONCEPT OF MARITIME DOMAIN AWARENESS

MDA is described as “the effective understanding of anything associated with the global maritime domain that could impact the security, safety, economy, or environment of a nation” [10]. In the past two decades, the primary focus has been towards terrorism, piracy, drug trafficking, illegal arms and ammunitions transport, smuggling, and illegal migration in the maritime domain [13].

In 2007, Chief of Naval Operations Admiral Michael Mullen signed a memorandum which provided an outline for the fleet MDA concept of operations

(CONOPS) in association with external MDA efforts [15]. In the course of implementing these guidelines, much work has already been done. Nevertheless, there is still much work to be done in this field. Because of a significant increase in maritime traffic, ever increasing dependence of commercial trade on the maritime domain, and a paradigm shift in the global security situation, there is now an even more urgent need to undertake autonomous MDA. This involves collecting massive amounts of global maritime data from various sources and agencies followed by synchronization and analysis of the data to discriminate between ‘normal’ and ‘abnormal’ behaviors [13].

B. ANOMALY DETECTION

An anomaly is sometimes referred to as ‘abnormal’ behavior and, in the literature, is also termed as “unusual” or “an outlier” [16]. In the maritime domain literature, there are different definitions for abnormal behavior [16]. Herein, abnormal ship movement is taken to mean movement of ships which are unlikely compared to most maritime traffic.

Anomaly detection is a stepwise process in which the first step is to define normalcy or standard behavior based on some past data. The next step is to compare each vessel’s present data with the pre-defined normalcy standards, followed by identifying anomalies. In the maritime domain, there can be several different types of anomalies. For instance, a stationary ship in the middle of a sea lane where most of the ships are moving may appear as an anomaly. The stationary ship, however, could be a survey vessel performing a seabed survey or a dredging vessel. Similarly, putting a lower or upper threshold on the speed of vessel for anomaly detection may classify slow-moving towing vessels or high-speed boats as being anomalous. Because several different types of simultaneous anomalies will always exist in the maritime domain, they need to be distinguished by different classifiers. The final goal is to compile and analyze the output of all the anomaly classifiers for subsequent generation of automated alerts, followed by human intervention, if necessary.

Much work has already been done in this field, and this thesis is intended to improve the ongoing research. Automated anomaly detection is believed to improve and enhance early warning, situational awareness, manpower usage, and decision support systems with less human intervention.

Anomaly detection is a very generic term broadly referred to as ‘abnormality detection’ in the maritime domain. In most cases, these abnormalities or anomalies are related to the kinematics of a vessel and may be further classified into various types. Different types of anomalies in the maritime domain are shown in Figure 2.

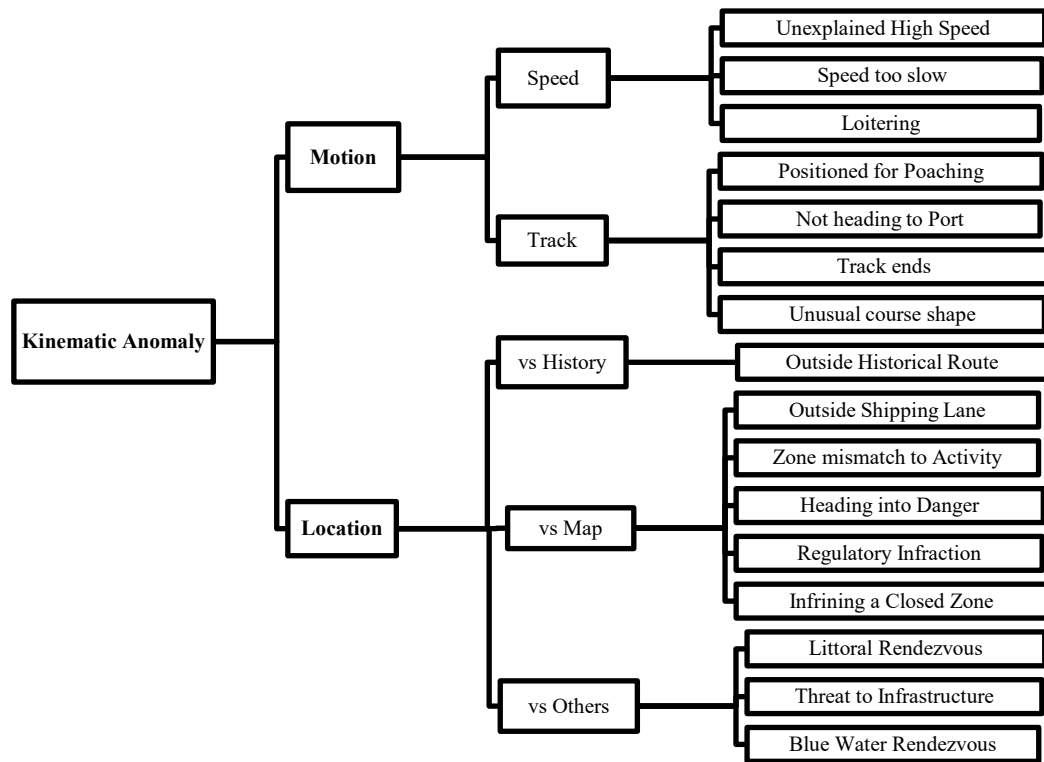


Figure 2. Types of Anomalies. Source: [16].

To detect each type of anomaly, there may be different algorithms and techniques, and it is very likely that a specific type of anomaly detection technique may work well for one but not so well for another. To obtain the most useful results, we use specific algorithms based on the type of anomaly. For the scope of this thesis, we restrict ourselves to anomalies where a vessel is not following a sea lane and where two or more ships are following one another.

The process of anomaly detection in the maritime domain mostly involves finding the outliers, which is typically depicted by abnormal movements during a ship’s voyage.

While most maritime traffic shows normal behavior, careful analysis of the vessels' behavior may be used to find abnormal or anomalous behavior. During the past few decades, machine learning has been heavily used in the research for finding outliers and abnormal behavior. Machine learning employs various algorithms depending the application and data availability.

Machine learning can be broadly classified as supervised or unsupervised machine learning. Supervised learning is a machine learning approach for determining a function from labeled training data. It examines the training data and derives a classifier, which can be used for mapping new examples. Some of the most common supervised learning algorithms are neural networks, support vector machines, and random forests. In unsupervised learning, labeled data is not available, and this approach learns about the hidden structure from input data. Depending on the application, various unsupervised learning approaches may be employed including clustering, Gaussian mixture models, and hidden Markov models [17].

C. AIS DATA

The Automatic Identification System, or AIS, is a maritime navigation safety communication standard from the International Telecommunication Union (ITU) adopted by the International Maritime Organization (IMO). It provides a vessel's identity, type, position, speed, course, destination, and navigational status automatically to shore stations and other vessels using the maritime Very High Frequency (VHF) band [18]. AIS is now required by IMO regulations on all passenger vessels of more than 150 gross tons displacement or registered to carry more than 150 passengers, all tankers, all vessels of more than 300 gross tons displacement, and all self-propelled boats over 65 feet in length [2], [18].

There are 27 different types of messages [19] that can be transmitted by AIS transceivers every two to ten seconds by ships at an estimated isotropic power level of 12.5 W [20]. For this thesis, we are using synthetic data similar to position report AIS message data.

D. PREPROCESSING OF DATA

Classification of ships into sea lanes and anomaly detection algorithms used in this thesis rely on consistent position data of ships. Although AIS could be used for this thesis, this poses a challenge of dealing with inconsistencies, including when any ship stops sending its location data, perhaps because of illicit activities, power outage, etc[9]. To mitigate this challenge, we use synthetic position data in the Cartesian coordinate system, and the use of actual AIS data is left for future work. Each data point may be considered similar to geodetic data of ships, which is obtained through AIS transmissions from each vessel. Since we are only using the position and time data, other fields of AIS data are not discussed in this thesis and are left for future study.

E. K-MEANS CLUSTERING

In this thesis, we use the k -means clustering algorithm, which is one type of unsupervised machine learning algorithm, for anomaly detection in the maritime domain. In this algorithm, all the observations are clustered into one of k clusters. The observations are clustered such that they belong to the cluster to which they have the greatest similarity. This is determined by proximity to the closest or nearest cluster center. The cluster center or centroid is the mean of the cluster members; hence, the mean serves as the defining factor of that cluster.

The k -mean algorithm assigns a set of n observations $\mathbf{x}_i = (\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, \dots, \mathbf{x}_n)$ into separate sets c_j of k clusters $c_j = \{c_1, c_2, c_3, \dots, c_k\}$, where each c_j is represented by a mean or centroid $\boldsymbol{\mu}_j$ [21]. The k -means clustering algorithm starts by selecting k centroids which, for our implementation, are randomly selected from the data. The flowchart for the algorithm is shown in Figure 3.

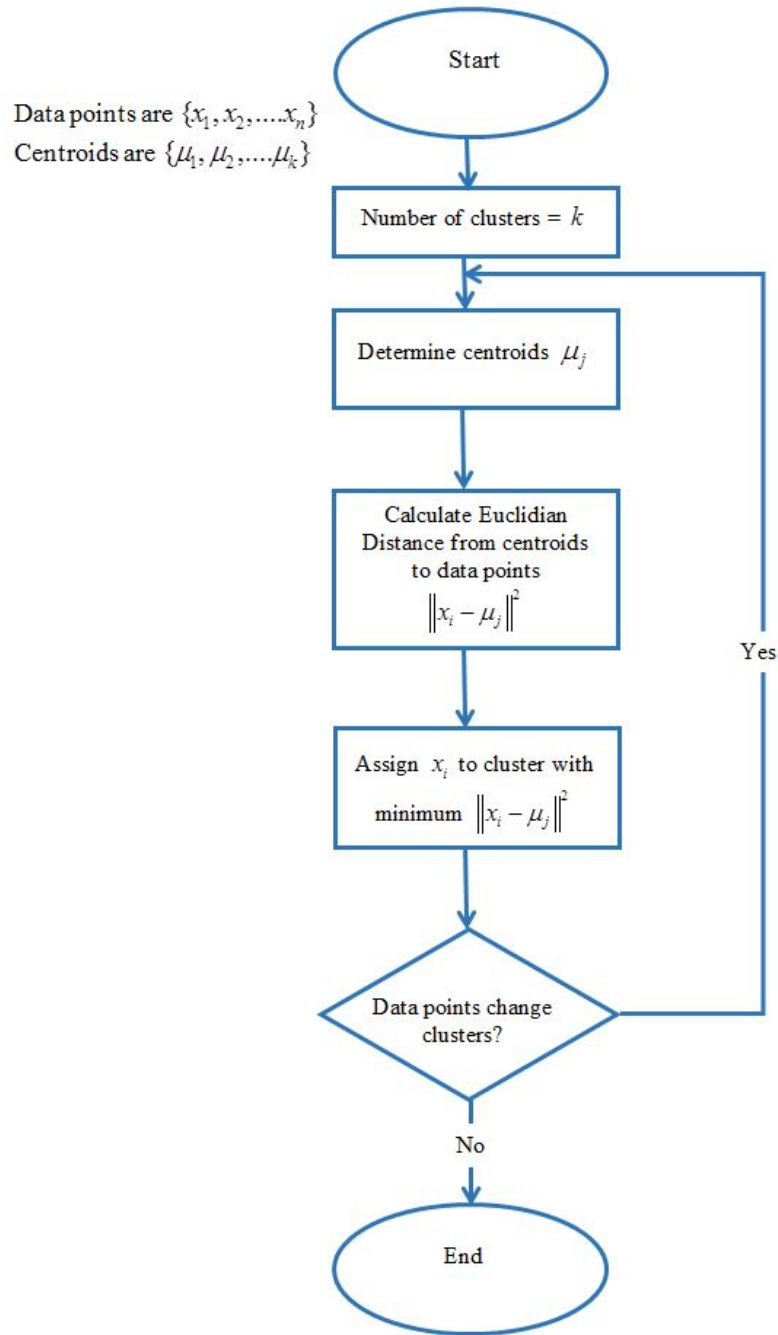


Figure 3. k -means Clustering Algorithm Flowchart

As shown in Figure 3, the algorithm iterates between the following steps:

1. Perform Cluster Assignment

Each of the n observations is assigned to one of the k clusters based on the minimum-squared Euclidian distance from centroid of each cluster $\boldsymbol{\mu}_j$ to the observation.

2. Update the Centroid

Each centroid $\boldsymbol{\mu}_j$ is updated as the observations assigned to each cluster c_j changes based on step 1 and is represented as

$$\boldsymbol{\mu}_j = \frac{1}{N_{c_j}} \sum_{i=1}^{N_{c_j}} \mathbf{x}_i \quad (1)$$

where the \mathbf{x}_i are the members of cluster j , and N_{c_j} is the number of observations in the cluster j . The same process is repeated in a loop multiple times, and the centroids change their location during each step until the centroids no longer move. The algorithm tries to minimize the objective function, which in this case is a squared-error function. The objective function is [9], [21]

$$J = \arg \min_c \sum_{j=1}^k \sum_{x_i \in c_j} \left\| \mathbf{x}_i - \boldsymbol{\mu}_j \right\|^2, \quad (2)$$

where $\left\| \mathbf{x}_i - \boldsymbol{\mu}_j \right\|^2$ is the Euclidian distance between \mathbf{x}_i and the centroid of cluster j .

The k -means algorithm is very sensitive to initially selected centroids, which in most cases are selected randomly and may, therefore, give a local objective minimum rather than a global minimum. This may be avoided by reiterating the algorithm various times.

In a very basic example of the k -means algorithm, seven individual data points having two variable scores A and B , shown in Table 1 and Figure 4, are to be grouped into two clusters.

Table 1. Data Points to be Grouped Using k -means Algorithm

Data Points	Score A	Score B
1	1.0	1.1
2	1.4	2.1
3	3.1	4.1
4	5.0	7.0
5	3.4	5.0
6	4.4	5.1
7	3.6	4.6

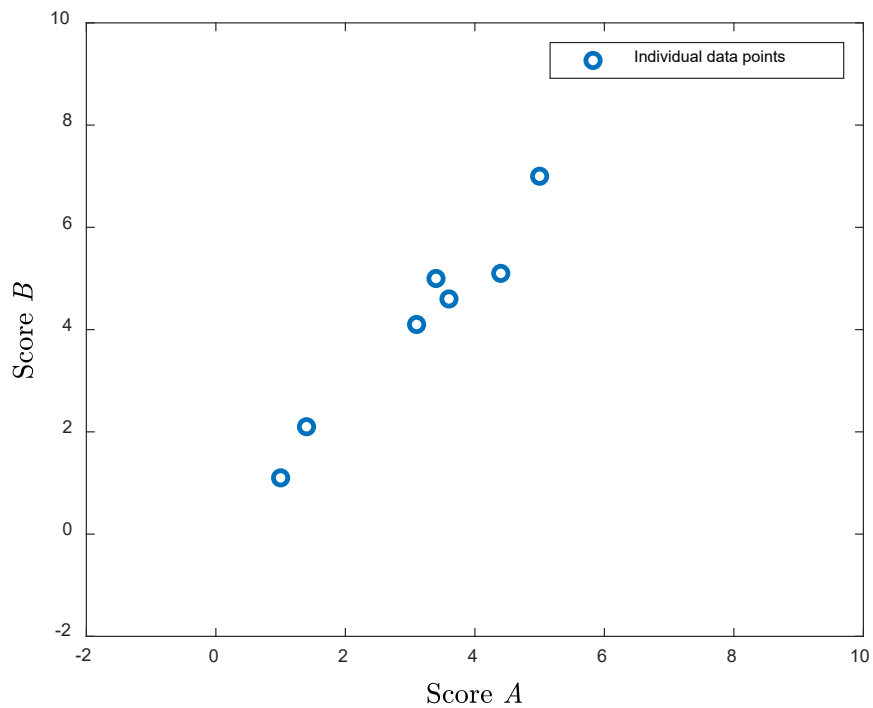


Figure 4. Data Points of Individual Scores

As the first step, two centroids are selected from the dataset. Commonly, the most distant points are chosen as centroids (in this case, (1.0, 1.1) and (5.0, 7.0), as shown in

Figure 5 and Table 2), and all the individuals are put into either of the two clusters based on the square of the Euclidian distance.

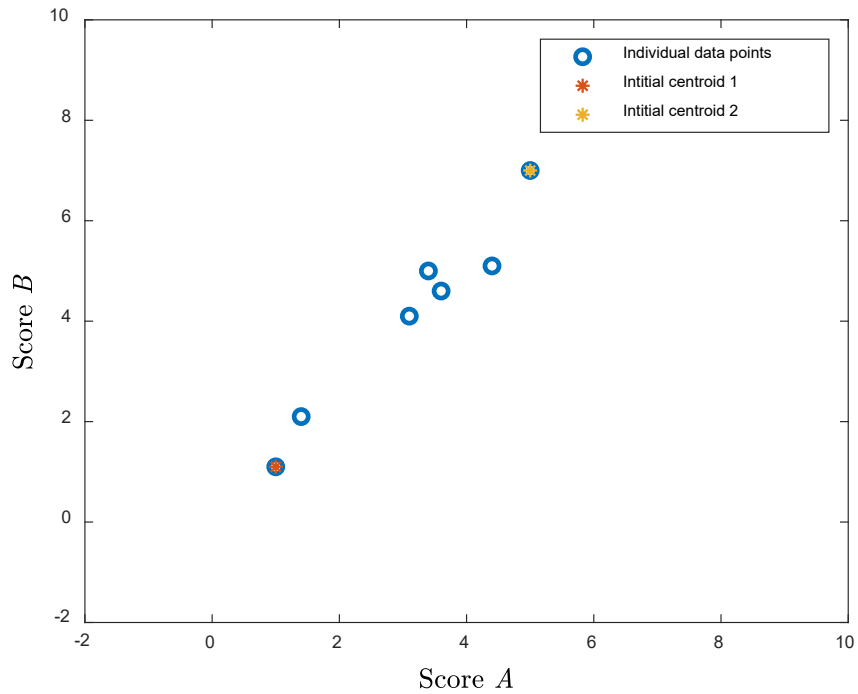


Figure 5. Data Points with Initially Selected Centroids

Table 2. Centroids of Data Points

	Data Points	Centroid
Cluster 1	1	(1.0,1.1)
Cluster 2	4	(5.0,7.0)

The remaining data points are then clustered into either of these two clusters based on their Euclidian distance from each centroid, and the centroid is also re-calculated each time a new data point is added to that cluster, as shown in Table 3.

Table 3. Data Points with New Centroids

Step Number	Cluster 1		Cluster 2	
	Data Points	New Centroid	Data Points	New Centroid
1	1	(1.0,1.1)	4	(5.0,7.0)
2	1,2	(1.2,1.6)	4	(5.0,7.0)
3	1,2,3	(1.8,2.4)	4	(5.0,7.0)
4	1,2,3	(1.8,2.4)	4,5	(4.2,6.0)
5	1,2,3	(1.8,2.4)	4,5,6	(4.2,5.7)
6	1,2,3	(1.8,2.4)	4,5,6,7	(4.1,5.4)

Since all the calculations are based upon the initial selection of the centroids, we need to recheck each data point to ensure it has been clustered in the right one. To accomplish this, we compare the Euclidian distance of each data point with both centroids, as shown in Table 4.

Table 4. Euclidian Distance of Each Data Point from Centroid

Data Point	Euclidian Distance from centroid of cluster 1	Euclidian Distance from centroid of cluster 2
1	1.5	5.3
2	0.5	4.2
3	2.1	1.6
4	5.6	1.8
5	3.0	0.8
6	3.7	0.42
7	2.84	0.9

Since data point 3 has minimum Euclidian distance from the centroid of cluster 2 rather than the centroid of cluster 1, it is clustered into cluster 2, and the centroid of cluster 2 is also updated. The final clusters are shown in Table 5 and Figure 6.

Table 5. Final Clusters Using k -means Algorithm

	Data points	Centroid
Cluster 1	1,2	(1.8,2.4)
Cluster 2	3,4,5,6,7	(3.9,5.1)

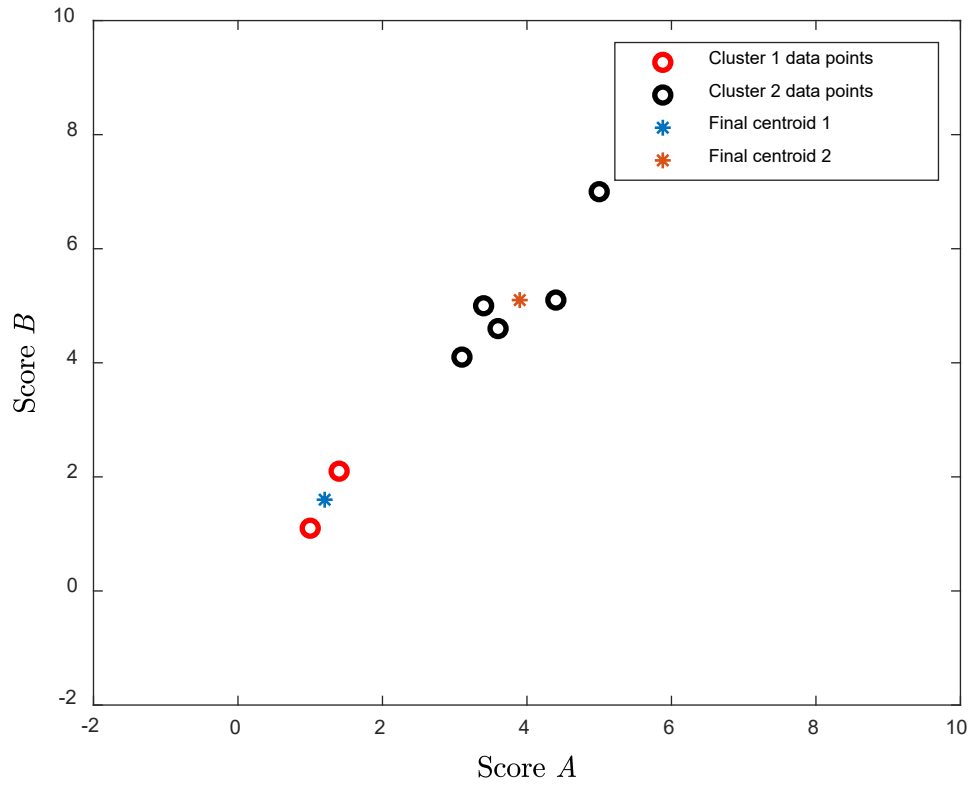


Figure 6. Final Clusters Using k -means Algorithm

THIS PAGE INTENTIONALLY LEFT BLANK

III. ALGORITHMS AND METHODS FOR CLASSIFICATION AND ANOMALY DETECTION

In this thesis, we analyze the problem of anomaly detection in MDA unlike the way most previous research has been done. Most of the previous work quantifies the difference between two tracks using Euclidian distance between position reports [9], [22]. As we show herein, this is suboptimal because we desire a distance metric that is zero when comparing two identical paths. Two ships may transmit different position reports even though they are underway on the exact same path. This scenario is presented in Figure 6, in which two ships transmit two different positional reports even though they are travelling on the exact same path. We suggest that the best metrics and algorithms will recognize the paths of these two ships as identical. Unfortunately, this is not true in the research in the literature to date [9], [22], [23].

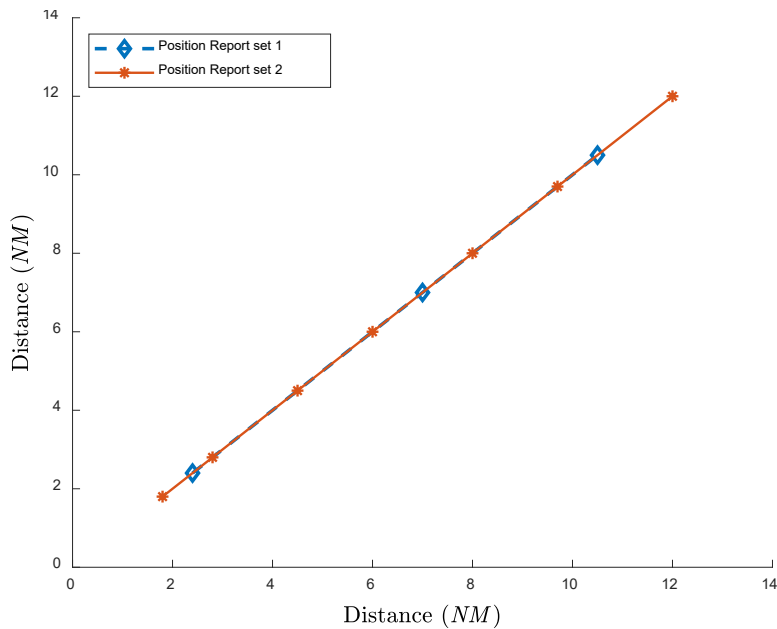


Figure 7. Positional Reports of Two Ships on Same Path

Since most of the machine learning algorithms and techniques used to classify ships' tracks into sea lanes, as well as anomaly detection, are data sensitive, both the

positional report sets shown in Figure 7, being very different from one another, yield different results even though they belong to the same straight line representing a sea lane. For this reason, in this thesis we aim to find a way to classify ships tracks into sea lanes and for anomaly detection that is not data sensitive. We designed a metric for comparing tracks that is much less sensitive to the specific location points and indeed measures the two superimposed tracks depicted in Figure 7 as identical, as desired. We devised an algorithm to calculate the geographical area between ships' tracks and use that area information to compare two tracks, define sea lanes, classify ships into sea lanes, and to identify anomalous behavior.

A. AREA METRIC

In this thesis, we design an area metric or distance function that defines the distance between each pair of elements of different ships' tracks by calculating the geographical area between them. By definition, a metric on a set X is a distance function, such that [24], [25]:

$$d : X \times X \rightarrow \mathbb{R}^+, \quad (3)$$

where \mathbb{R}^+ is the set of non-negative real numbers, and for all $\mathbf{x}, \mathbf{y}, \mathbf{z} \in X$, the following conditions are satisfied:

$$d(\mathbf{x}, \mathbf{y}) \geq 0, \quad (4)$$

$$d(\mathbf{x}, \mathbf{y}) = 0 \Leftrightarrow \mathbf{x} = \mathbf{y}, \quad (5)$$

$$d(\mathbf{x}, \mathbf{y}) = d(\mathbf{y}, \mathbf{x}), \quad (6)$$

and

$$d(\mathbf{x}, \mathbf{z}) \leq d(\mathbf{x}, \mathbf{y}) + d(\mathbf{y}, \mathbf{z}). \quad (7)$$

In this thesis, we calculate geographical area between two ships' tracks using the area calculation algorithm, which is discussed in detail in the ensuing paragraphs, and we

refer to this area as the “area metric.” We represent a ship’s track \mathbf{t} as a $2 \times N$ matrix as:

$$\mathbf{t} = \begin{bmatrix} x_1 & x_2 & x_3 & x_4 & \dots & x_N \\ y_1 & y_2 & y_3 & y_4 & \dots & y_N \end{bmatrix} \quad (8)$$

where each column indicates a data point or position report of the ship’s track in temporal order.

To compute the area metric, we find the geographical area between two ships’ tracks by the sum of triangular areas formed by the adjacent points of the sea lane and the ship’s track represented in Figure 8.

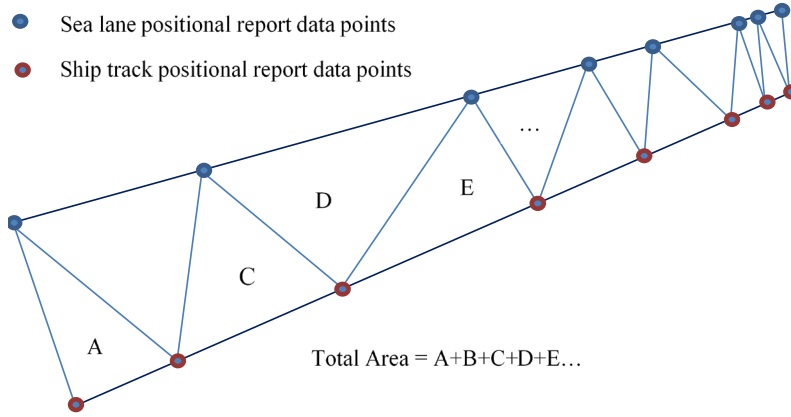


Figure 8. Area between Sea Lane and Ship’s Track

The area of each triangle, shown in Figure 8, is calculated separately using Heron’s formula [26]

$$\text{Area} = \sqrt{s(s - d_{12})(s - d_{23})(s - d_{31})} \quad (9)$$

where $s = 0.5(d_{12} + d_{23} + d_{31})$ and d_{12} , d_{23} and d_{31} are the Euclidian distances shown in Figure 9.

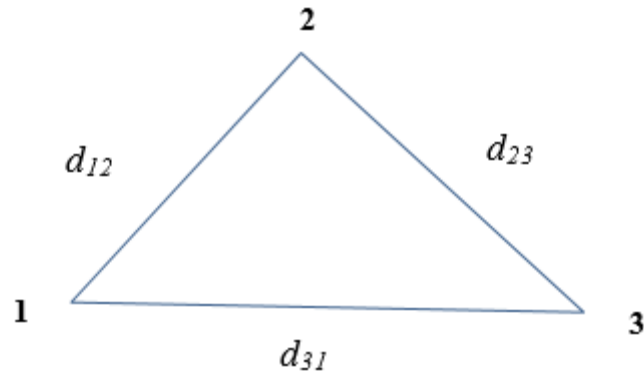


Figure 9. A Triangle Represented by Three Points

The Euclidian distance [27] between two points, illustrated in Figure 10, is

$$d_{12} = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \quad (10)$$

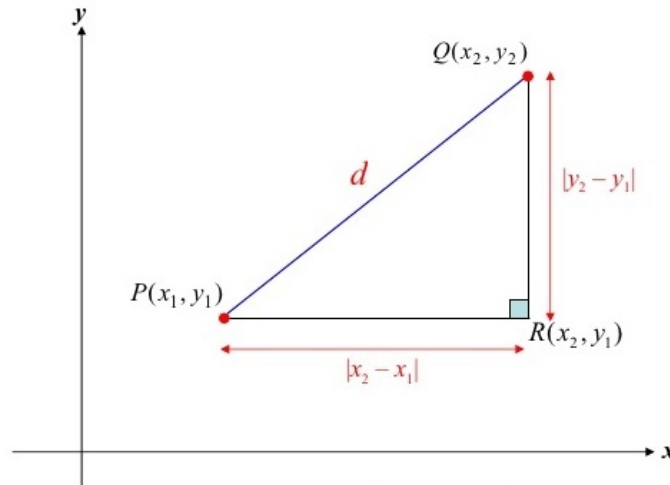


Figure 10. Euclidian Distance d between Points P and Q

The sum of all the triangular areas labeled A, B, C shown in Figure 8 gives the complete area between the sea lane and the vessel's track. If the vessel stays close to the defined sea lane, the area is minimal, and in cases where either it stays far away from the sea lane or does haphazard movement, it has a large area with respect to the selected sea lane. This algorithm has an advantage over most of the previous research in this area

because the algorithm is not sensitive to the frequency of position reports of ships. Moreover, if two ships follow the same track but have different sets of position reports, as shown in Figure 7, they will have a zero area metric and, therefore, are classified as identical tracks.

The same algorithm for area calculation can be used to calculate the area between any shape of sea lane and ship's track. A graphical example of the area calculation is shown in Figure 11. The total area between the ship's track and the sea lane is the sum of all 15 triangular areas.

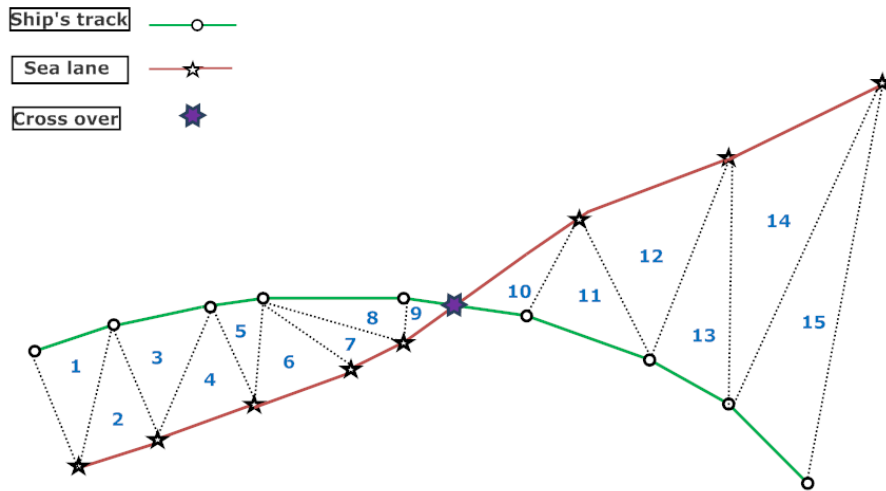


Figure 11. Area between Ship's Track and Sea Lane

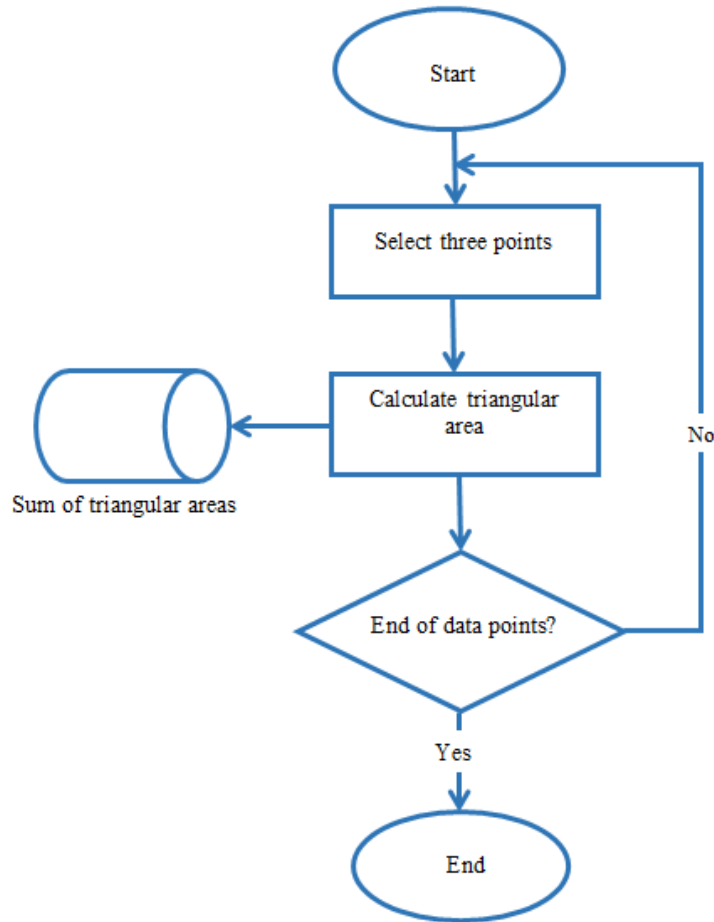


Figure 12. Algorithm for Calculating Area Metric

The algorithm is as follows:

1. Initially two data points, one each from the sea lane and the ship's track, are selected.
2. Existence of crossover is checked each time before selecting a third data point. A crossover is illustrated using the purple six-pointed star in Figure 11.
3. If no crossover exists, the third point is selected based on the minimum Euclidian distance from the mean of the initial data points of sea lane and ship's track to form the triangular region for subsequent area calculation.

4. In the case where a crossover point exists before the next data points of either sea lane or ship's track, as shown in Figure 13, the crossover point is calculated using

$$m_1 = \frac{y_{12} - y_{11}}{x_{12} - x_{11}} , m_2 = \frac{y_{22} - y_{21}}{x_{22} - x_{21}} , \quad (11)$$

$$b_1 = y_{11} - m_1 x_{11} , b_2 = y_{21} - m_2 x_{21} , \quad (12)$$

and

$$(x_c, y_c) = \left(\frac{b_2 - b_1}{m_1 - m_2}, m_1 x_c + b_1 \right) . \quad (13)$$

The crossover point is then taken as the third point to complete the triangular area.

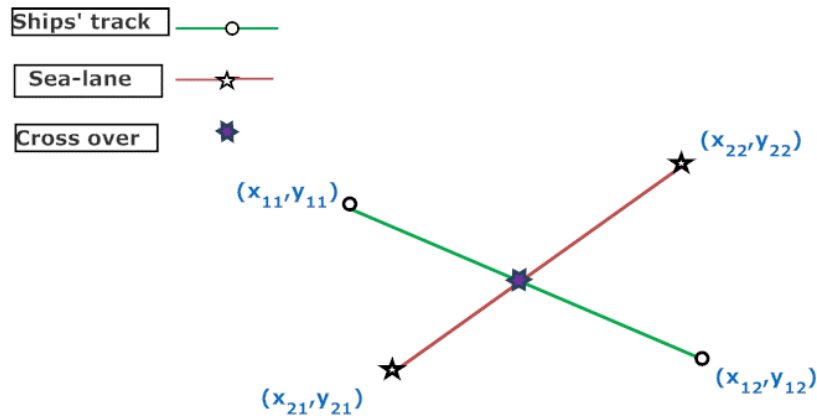


Figure 13. Crossover of a Sea Lane and a Ship's Track

5. After calculation of the first triangular area, the same process is repeated until all the triangular areas shown in Figure 11 are calculated and summed to yield the total area between the sea lane and the ship's track.

B. DEFINING A SEA LANE USING AREA METRIC AND MODIFIED K-MEANS CLUSTERING ALGORITHM

The sea lanes are defined in this thesis using historic data of ships' tracks. The algorithm is inspired by the k -means clustering algorithm and is, therefore, referred to as the modified k -means clustering algorithm in this thesis. The k -means algorithm groups data into clusters based on their Euclidian distance from the mean of the cluster. In this thesis our modified k -means algorithm clusters ship's tracks, which are sets of matrices based on their areas with respect to other sea lanes using the area metric discussed in Chapter III, Section A. The flowchart for defining sea lanes using historic ships' track data is shown in Figure 14. The algorithm consists of the following steps.

1. Determining Number of Sea Lanes to Be Defined

For this thesis, we assume that we already know the number of sea lanes to be defined in a particular geographical area in the maritime domain and any technique for investigating the number of sea lanes is left for the future work.

2. Finding Areas between Ships' Tracks

Using the area metric defined in Section A of this chapter, we find the area between each ship's track and all the other tracks. Suppose we have n number of ships' tracks, we find the area between the first ship's track and all the other $(n-1)$ tracks. A similar step is followed by changing the reference ship's track and finding the area with respect to all the other ships' tracks, giving a total of $0.5n(n-1)$ distinct areas. For the purpose of explanation, we set the number of ships' tracks as $n=5$ in a sea lane. So there are $0.5n(n-1)=10$ distinct non-zero areas between ships' tracks as shown in Table 6, and they are represented by $d(\mathbf{t}_i, \mathbf{t}_j)$ where $i = j \in \{1, \dots, 5\}$ and $i \neq j$. Notice that the area $d(\mathbf{t}_i, \mathbf{t}_j) = 0$ when $i = j$ as the area of any ship's track in reference to itself is zero, satisfying (5). Also notice that $d(\mathbf{t}_x, \mathbf{t}_y) = d(\mathbf{t}_y, \mathbf{t}_x)$ as the area between two ships' tracks is always equal, satisfying (6).

Ship tracks are $\{t_1, t_2, \dots, t_{N_t}\}$ where N_t is the total number of ship tracks
 Sea lanes are $\{s_1, s_2, \dots, s_{N_s}\}$ where N_s is the total number of sea lanes

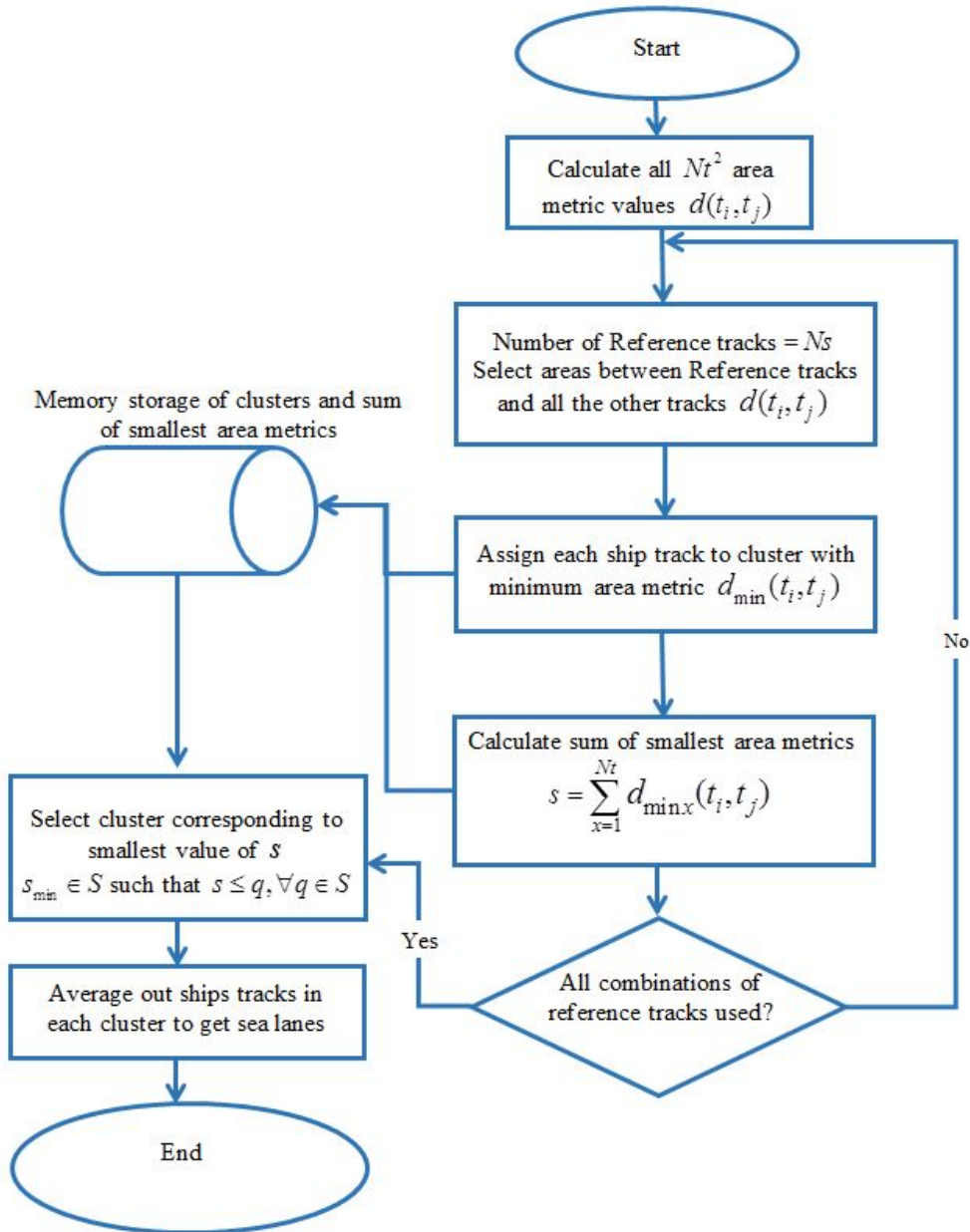


Figure 14. Flowchart for Defining Sea Lane from Ships' Tracks

Table 6. Example of Areas between Ships' Tracks (in NM^2)

		Track 1	Track 2	Track 3	Track 4	Track 5
		$d(\mathbf{t}_i, \mathbf{t}_1)$	$d(\mathbf{t}_i, \mathbf{t}_2)$	$d(\mathbf{t}_i, \mathbf{t}_3)$	$d(\mathbf{t}_i, \mathbf{t}_4)$	$d(\mathbf{t}_i, \mathbf{t}_5)$
Track 1	$d(\mathbf{t}_1, \mathbf{t}_j)$	0	5	7	2	3
Track 2	$d(\mathbf{t}_2, \mathbf{t}_j)$	5	0	6	3	4
Track 3	$d(\mathbf{t}_3, \mathbf{t}_j)$	7	6	0	4	6
Track 4	$d(\mathbf{t}_4, \mathbf{t}_j)$	2	3	4	0	7
Track 5	$d(\mathbf{t}_5, \mathbf{t}_j)$	3	4	6	7	0

3. Clustering the Ships' Tracks Using Modified k -Means Algorithm

These areas are then clustered into k sea lanes, which we are interested in determining. In this thesis, we are not using the actual k -means algorithm as discussed in the previous chapter since the ships' tracks are matrices, and each area with respect to another ship's tracks is a scalar quantity. Following are the steps involved in our algorithm. For example and explanation purposes, we cluster $n = 5$ ships' tracks into $k = 2$ sea lanes in the following steps:

1. Since we have already calculated area $d(\mathbf{t}_i, \mathbf{t}_j)$ for all ships' track pairs, as shown in Table 6, we pick two reference tracks at a time and take the area between these two tracks and all the other tracks, i.e., $d(\mathbf{t}_i, \mathbf{t}_j)$ where $i \in \{1, 2\}$ and $j \in \{1, \dots, 5\}$.
2. Assign each ship's track to a cluster (sea lane), which yields minimum area $\min_i [d(\mathbf{t}_i, \mathbf{t}_j)]$, with respect to the two reference tracks. Cluster assignment for the first two reference ships' tracks are shown in Table 7.

Table 7. Cluster Assignment Based on Minimum Area Metric

		Track 1	Track 2	Track 3	Track 4	Track 5
		$d(\mathbf{t}_i, \mathbf{t}_1)$	$d(\mathbf{t}_i, \mathbf{t}_2)$	$d(\mathbf{t}_i, \mathbf{t}_3)$	$d(\mathbf{t}_i, \mathbf{t}_4)$	$d(\mathbf{t}_i, \mathbf{t}_5)$
Reference Track 1	$d(\mathbf{t}_1, \mathbf{t}_j)$	0	5	7	2	3
Reference Track 2	$d(\mathbf{t}_2, \mathbf{t}_j)$	5	0	6	3	4
Cluster Assignment based on minimum area metric		1	2	2	1	1
Smallest area metric $\min_i [d(\mathbf{t}_i, \mathbf{t}_j)]$		0	0	6	2	3
Sum of smallest area metrics $s = \sum_{j=1}^5 \min_i [d(\mathbf{t}_i, \mathbf{t}_j)]$						11

3. Take the sum of minimum areas $s = \sum_{j=1}^5 \min_i [d(\mathbf{t}_i, \mathbf{t}_j)]$ as shown in Table 7 and store it in memory with the cluster assignment of ships' tracks.
4. Repeat steps 1–3 for all possible combinations of reference tracks. Using the binomial coefficient, we have a total of ten distinct combinations of reference ships' tracks,

$$C(n, r) = \frac{n!}{r!(n-r)!} \quad (14)$$

where $n = 5$ and $r = 2$ in this example. For each combination of reference tracks, we record the corresponding values of the sum of minimum areas s in a matrix S such that $S = [s_1, s_2, s_3, \dots, s_{10}]$ and the cluster assignments of ships' tracks.

5. The cluster corresponding to the smallest value of the sum of the minimum areas, i.e., $s_{\min} \in S$ such that $s \leq q, \forall q \in S$, is the best cluster for the ships' tracks.
6. Based on these clusters, we average the ships' tracks in each cluster, and this average defines the sea lane.
7. For this thesis, we assumed that all the ships' tracks have the same number of data points, making the process of averaging relatively simple. If actual AIS data is used, the process of averaging the ships' tracks might be a challenge given that all ship's tracks have a different number of data points. For this thesis, we did not take this problem into account and have left this as future work.

C. CLASSIFICATION OF SHIPS' TRACKS BY SEA LANE

Classification of ships' tracks by sea lane is an important part of MDA and provides an insightful perspective on the anomaly detection problem. In this thesis, we classify ships by sea lane using the area calculation algorithm already discussed and then apply other techniques to the areas of each ship's track with respect to each sea lane. These techniques are discussed in the following sections.

1. User-Defined Threshold

Ships' tracks can be classified into different sea lanes based on a user-defined threshold for their respective areas with different sea lanes. These area thresholds may be different for each sea lane or sections of sea lane depending on their geographical location. For example, they may be small for littoral waters where ships normally tend to stay close to the respective sea lanes or may be large for open sea as they may deviate considerably from sea lanes based on navigational situations.

2. The Modified k -means Clustering Algorithm

The modified k -means algorithm defined in the previous section can also be used to classify ships into sea lanes using their areas with respect to each sea lane. A flowchart for classifying ships' tracks into known sea lanes is shown in Figure 15.

Ship tracks are $\{t_1, t_2, \dots, t_{Nt}\}$ where Nt is the total number of ship tracks
Sea lanes are $\{s_1, s_2, \dots, s_{Ns}\}$ where Ns is the total number of sea lanes

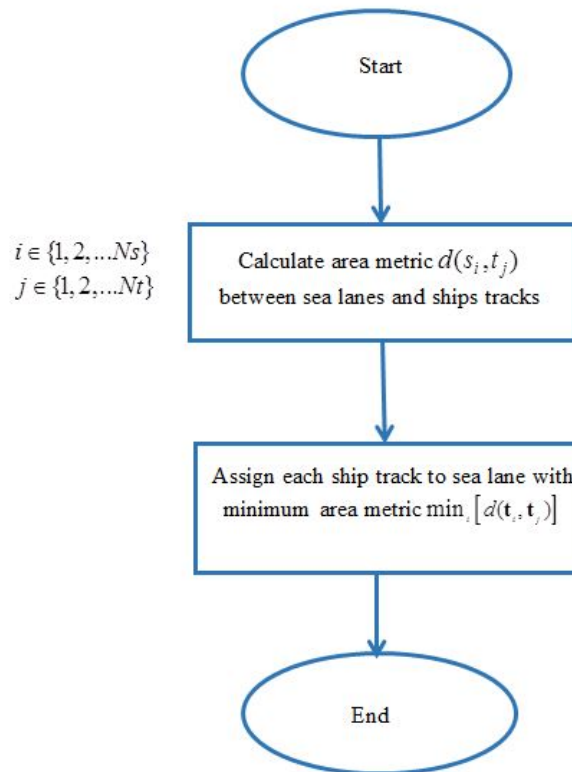


Figure 15. Flowchart for Classifying Ships' Tracks into Sea Lanes

The algorithm consists of the following steps:

1. As we have already described the algorithm for defining sea lanes in Chapter III, Section B, for this section we assume that we know the sea lanes. We calculate the area metric between each sea lane and each ship's

track. If there are two sea lanes, i.e., $i \in \{1, 2\}$ and five ships' tracks, i.e., $j \in \{1, 2, \dots, 5\}$, the number of total area metric values, $d(\mathbf{s}_i, \mathbf{t}_j)$, is ten.

2. Assign each ship's track to a cluster (sea lane) that has minimum area $\min_i [d(\mathbf{t}_i, \mathbf{t}_j)]$ between itself and sea lane. The sea lane can be considered as a centroid of the cluster, and the area between sea lane and a ship's track is analogous to the Euclidian distance between the centroid and data points in the conventional k -means algorithm.

3. Bayes' Theorem

As mentioned earlier in this thesis, the areas of all the ships' tracks are calculated with each sea lane and then, using a modified k -means algorithm, these ships' tracks are classified by sea lane. Here we use the same classification information from Chapter III, Section C, and Subsection 2. Then using Bayes' theorem, we find the conditional probability of being in a specific sea lane given the areas with respect to all the sea lanes. Bayes' theorem states that the probability of event A given that event B has occurred is [28]

$$P(A|B) = \frac{P(AB)}{P(B)}, \quad (15)$$

where $P(A|B)$ is the conditional probability of A given B , $P(B)$ is the probability of B , and $P(AB)$ is the joint probability of A and B . This can also be graphically illustrated in the form of a Venn diagram as shown in Figure 16.

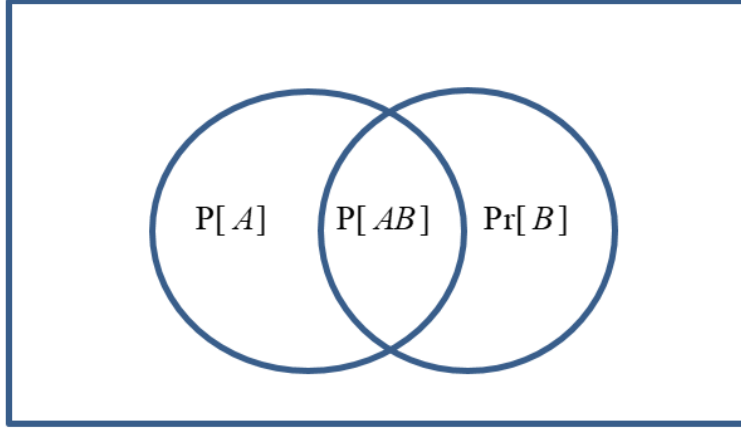


Figure 16. Venn Diagram for Bayes' Probabilities Theorem

The joint probability can, therefore, be expressed as

$$P(AB) = P(A | B) \text{Pr}(B) = P(B | A) \text{Pr}(A). \quad (16)$$

Hence, one conditional probability can be computed from another using [12]

$$P(A | B) = \frac{P(B | A) P(A)}{P(B)}. \quad (17)$$

For the purpose of illustration, consider two events A and B , where event A means the ship is in sea lane A and event B means the ship is in sea lane B . For this thesis, we assume A and B are exhaustive, i.e., $P(A \cup B) = 1$, and mutually exclusive, i.e., $P(AB) = 0$. This means that a ship belongs to either of the sea lanes A or B but not both. The area metric of a ship's track with respect to sea lanes A and B is represented by continuous random variables X and Y where they may take on any non-negative real value. Bayes' theorem for continuous random variables X , Y , and an event A is represented as [28]

$$P(A | X = x, Y = y) = \frac{f_{X,Y|A}(x,y) P(A)}{f_{X,Y}(x,y)}, \quad (18)$$

where $f_{X,Y|A}(x,y)$ is the conditional probability density function (PDF) for the area metric of ships' tracks with respect to both sea lanes given that the track is in sea lane A and $f_{X,Y}(x,y)$ is the joint PDF for ships' tracks. We then use the Law of Total Probability [28] to evaluate

$$f_{X,Y}(x,y) = f_{X,Y|A}(x,y)P(A) + f_{X,Y|B}(x,y)P(B). \quad (19)$$

Substituting (19) into (18) yields the conditional probabilities of event (sea lane) A and sea lane B , given ships' track area metrics:

$$P(A|X=x,Y=y) = \frac{f_{X,Y|A}(x,y)P(A)}{f_{X,Y|A}(x,y)P(A) + f_{X,Y|B}(x,y)P(B)}, \quad (20)$$

$$P(B|X=x,Y=y) = \frac{f_{X,Y|B}(x,y)P(B)}{f_{X,Y|A}(x,y)P(A) + f_{X,Y|B}(x,y)P(B)}. \quad (21)$$

As the area metric of any ship's track with respect to a sea lane is a continuous random variable, in cases where more than one sea lane exists, the area metrics of a ship's track forms a set of continuous random variables with each random variable equal to the area metric with respect to a particular sea lane. Since we assumed two sea lanes (A and B) in a particular maritime area, each ship's track has two random variables, X and Y . The random variables X and Y , being area metrics of a specific ship's track with respect to sea lanes A and B , respectively, can each take on any non-negative real value.

In this thesis, for convenience we assume the area metrics of any ship's track with respect to sea lanes A and B (i.e., random variables X and Y) are independent Gaussian random variables with $X \sim N(\mu_X \neq 0, \sigma_X^2)$ and $Y \sim N(\mu_Y \neq 0, \sigma_Y^2)$. The assumption that X and Y are Gaussian random variables is made for simplicity. In reality, the area metric between ships' tracks can never be negative, thus X and Y are not actually Gaussian random variables. The random variables X and Y could be modeled with other distribution types, e.g., lognormal or Poisson distribution. Nonetheless, this is left for future work.

In order to determine the probability of a ship's track being in sea lane A or sea lane B we must calculate the conditional PDFs of the area metrics given sea lane A and sea lane B ; i.e., $f_{X,Y|A}(x,y)$ and $f_{X,Y|B}(x,y)$. The expressions for these PDFs are

$$f_{X,Y|A}(x,y) = \frac{1}{\pi\sigma_{X|A}\sigma_{Y|A}} \exp\left(-0.5\left(\left(\frac{x-\mu_{X|A}}{\sigma_{X|A}}\right)^2 + \left(\frac{y-\mu_{Y|A}}{\sigma_{Y|A}}\right)^2\right)\right) \quad (22)$$

and

$$f_{X,Y|B}(x,y) = \frac{1}{\pi\sigma_{X|B}\sigma_{Y|B}} \exp\left(-0.5\left(\left(\frac{x-\mu_{X|B}}{\sigma_{X|B}}\right)^2 + \left(\frac{y-\mu_{Y|B}}{\sigma_{Y|B}}\right)^2\right)\right) \quad (23)$$

where $\mu_{X|A}$, $\mu_{X|B}$, $\mu_{Y|A}$, and $\mu_{Y|B}$ are conditional means of random variables X and Y , respectively, and $\sigma_{X|A}$, $\sigma_{X|B}$, $\sigma_{Y|A}$, and $\sigma_{Y|B}$ are conditional standard deviations. These parameters may be estimated from a historic training dataset of ships' tracks in a particular area after classification into sea lanes using a technique such as the one discussed in Chapter III, Section C, and Subsection 1 and 2. For conditional PDF $f_{X,Y|A}(x,y)$, we set $\mu_{X|A} = 0$ for convenience as all the ships' tracks belonging to sea lane A have a small area metric with respect to sea lane A . Similarly for $f_{X,Y|B}(x,y)$, we set $\mu_{Y|B} = 0$. We also needed to multiply the conditional PDFs in (22) and (23) by a factor of two because area metrics X and Y are assumed to be non-negative values, which reduces the area under the zero mean normal PDF by one half. Finally, the posterior probability of each ship's track being in sea lane A and sea lane B may be found using (20) and (21). Based on these conditional probabilities, we can classify ships' tracks into different sea lanes. The PDF for ships' tracks $f_{X,Y}(x,y)$ are shown in Figures 17 and 18 and are comprised of the two conditional PDF curves $f_{X,Y|A}(x,y)$ and $f_{X,Y|B}(x,y)$.

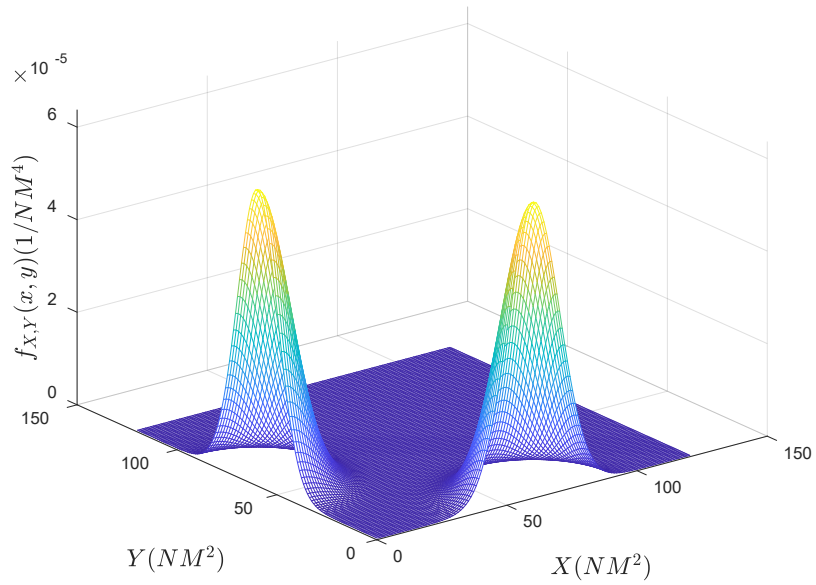


Figure 17. PDF for Ships' Tracks $f_{X,Y}(x,y)$ —Side View

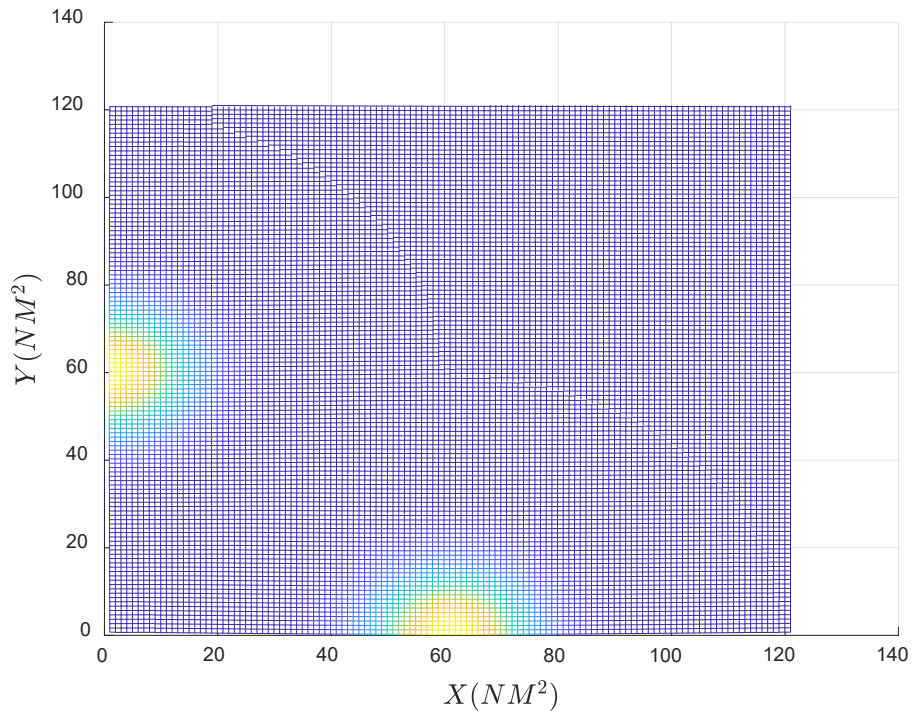


Figure 18. PDF for Ships' Tracks $f_{X,Y}(x,y)$ —Top View

D. ANOMALY DETECTION USING AREA METRIC

Anomaly detection is a very broad term, including many scenarios and situations as discussed in the previous chapter. For this thesis, we have tried to detect the following two types of anomalies:

1. Detection of outliers in standard sea lanes.
2. Detection of groups of ships following or paralleling in a sea lane.

The aforementioned anomalies may be detected using concepts based on those we have discussed for classifying ships' tracks into sea lanes. To do so, we first find the area between ships' tracks and sea lanes using the herein-developed area metric algorithm and then apply the modified k -means algorithm or a user-defined threshold.

1. Detecting Outliers in Standard Sea Lanes

We discussed the method of identifying sea lanes in Section B of this chapter. The maritime traffic in a specific sea lane normally stays close to the defined sea lane, and tracks of those ships have a small area with respect to the sea lane. In cases where a ship deviates from a standard sea lane, the area with respect to that specific sea lane will start to increase, and the probability of being in that specific sea lane is less.

For outlier detection, we divide the maritime domain into relatively small geographical areas and identify the numbers of sea lanes in that specific area. Based on the historic data of all the sea lanes in that defined area, we calculate the probability of each ship being in each sea lane using Bayes' theorem of conditional probabilities, as discussed in Section C of this chapter. Using these probabilities, we are able to extract information that is indicative of anomalous behavior.

2. Detecting Group of Ships Following or Parallel in a Sea Lane

If two or more ships are following one another or paralleling each other over a significant period of time, that may be considered anomalous behavior. In this thesis, to identify these ships' tracks we use a method similar to the one described in Section B of

this chapter. The phenomenon of detecting ships following each other in the maritime domain, however, not only requires analysis of the ships' spatial track data but also their temporal track data. To account for this, we convert the time information of each ship's track into distance information by multiplying by the speed of the ship. As a result, each ship's track has a three-dimensional column matrix, i.e., $\bar{X} = [\mathbf{x} \ \mathbf{y} \ \mathbf{t}]$, where \mathbf{x} and \mathbf{y} are column vectors representing spatial locality of ship's track and \mathbf{t} is the column vector representing its temporal locality in Greenwich Mean time (GMT) – 24-hour format converted to minutes. For example, if the time for the seventh position report is 0500, then $t_7 = 300$ minutes. To measure the similarity or distinction between ships' tracks, we use the area metric defined in Section A of this chapter by taking into account spatial as well as temporal track data. The steps to be followed for finding the ships' following one another are listed here and are also shown in a flowchart in Figure 19.

1. We find the areas between the ships' tracks using the area metric algorithm similar to one defined in Section A of this chapter. The only difference is that each ship's track is represented by three column vectors instead of two. The Euclidian distance between different points of a ship's track, required for calculating the area between ships' tracks as shown in Figure 11, not only accounts for spatial locality but also for the temporal locality of the track.
2. Next, we define a user-defined minimum area threshold $A_{threshold}$, which represents the upper limit of an anomaly. Any value of the area metric smaller than $A_{threshold}$ is considered anomalous, i.e., ships following one another.
3. For this analysis, we use the concept of a sliding window of a specific number of data points for calculating the area metric between ships' tracks. For example, we consider only the past 20 positional reports of all ships to calculate the area metric and, thus, identify ships following one another. The sliding window technique for position reports, however, may not work

well while using actual AIS data, since different ships transmit position reports with different frequencies (e.g., once every five minutes, once every minute, etc.). It may be worthwhile to explore other techniques of considering specific length (in nautical miles or time) for all ships' tracks, but this is relegated to future work.

Ship tracks are $\{t_1, t_2, \dots, t_{Nt}\}$ where Nt is the total number of ship tracks

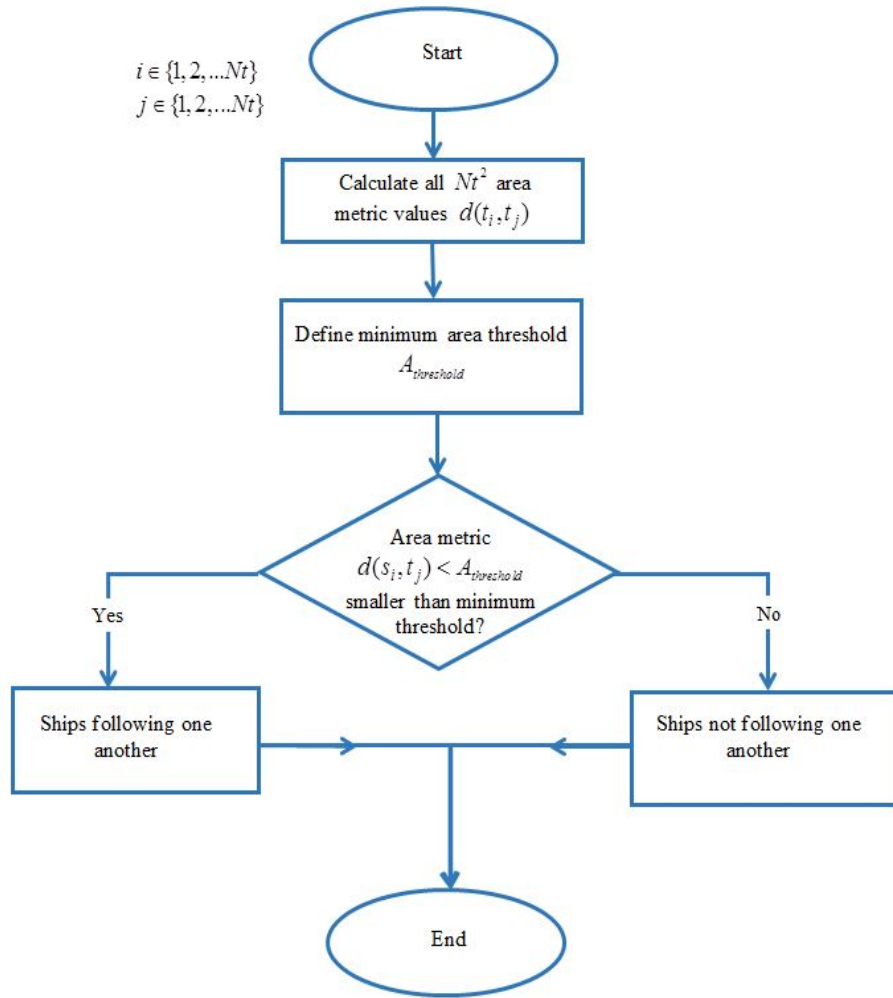


Figure 19. Flowchart for Anomaly Detection—Ships Following One Another in a Sea Lane

The algorithms discussed in this chapter will work with vessels' position report data obtained from different sensor, e.g., AIS, GPS, radar, etc. The algorithms are also less sensitive to the data inconsistencies, which is an important challenge being faced in the maritime domain. In this chapter, we discussed a method of defining sea lanes using historical ships' track data and employing the modified k -means algorithm, and classified ships' tracks into sea lanes using user-defined thresholds, a modified k -means algorithm, and the Bayes Theorem of conditional probabilities. We also discussed two approaches to detect anomalous vessel behavior in the maritime domain: ships not following any sea lanes and ships paralleling or following one another. All the aforementioned algorithms used an area metric algorithm between ships' tracks to measure the similarity or distinction between ships' tracks.

IV. RESULTS

The methods and algorithms discussed in Chapter III are implemented, verified, and the results analyzed in this chapter. As explained earlier, to clearly present the concept of this thesis, we work with synthetic data of ships' tracks rather than actual AIS data. The synthetic data comprises ship's track data in Cartesian coordinates instead of geographical coordinates. Implementation of this concept using actual AIS data is left for future work. All the concepts and algorithms of this thesis are implemented using the MATLAB programming language.

A. GENERATING SHIP'S TRACK DATA

For generating ships' tracks, we first define an imaginary sea lane comprised of data points. These data points may or may not be equidistant. For the purpose of explanation, we define a straight sea lane, going from Point A to Point B , and comprising ten data points, as shown in Figure 20.

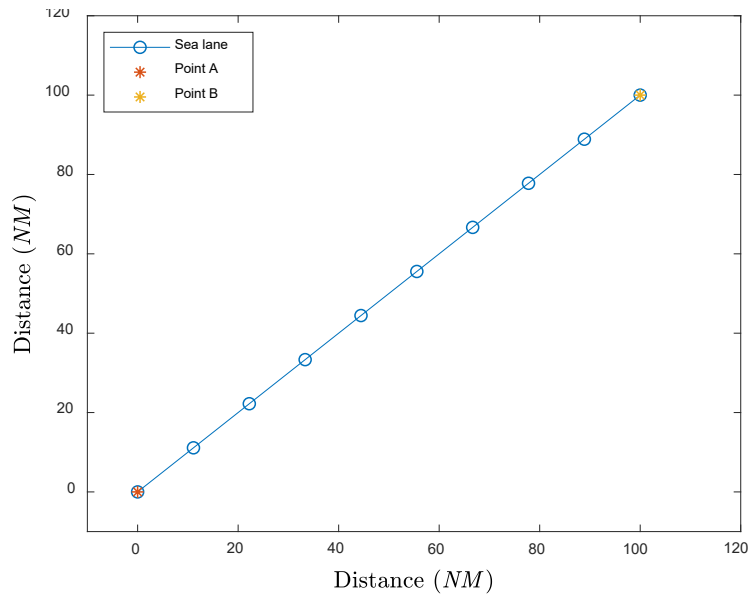


Figure 20. Sea Lane between Points A and B

All the ships' tracks in this sea lane are generated by adding colored random Gaussian noise to the sea lane data points. In Figure 21, we generated six ships' tracks by adding different levels of noise to the sea lane.

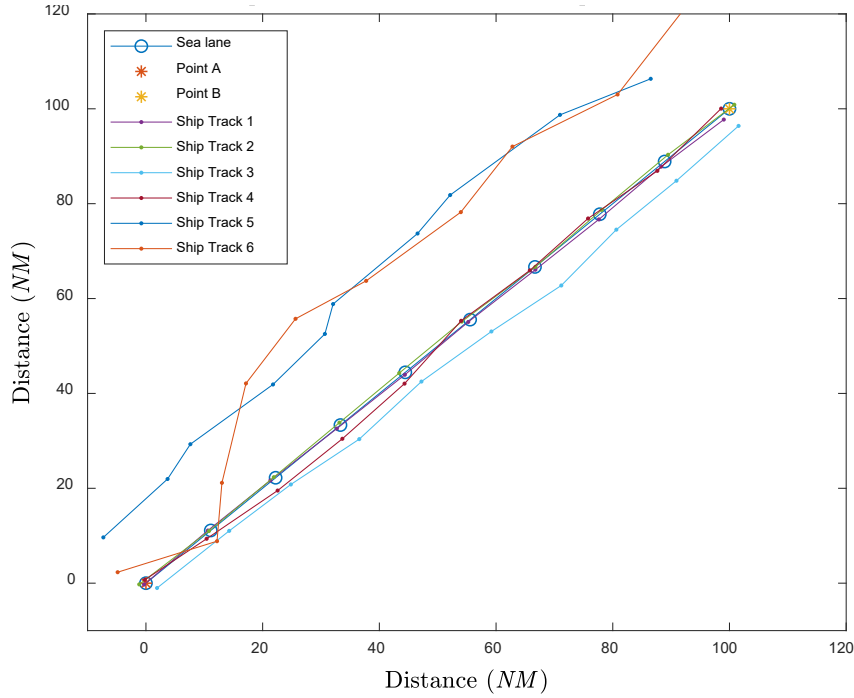


Figure 21. Ships' Tracks and a Sea Lane from Point A to B

B. AREA METRIC ALGORITHM BETWEEN SHIPS' TRACKS AND SEA LANE

The algorithm for calculating the area between a ship's track and a sea lane is explained in Chapter III, Section A. The algorithm was implemented in MATLAB code, which is included in the Appendix. The area metric for the six ships' tracks with respect to the sea lanes illustrated in Figure 21 are listed in Table 8.

Table 8. Area of Ships' Tracks with Respect to Sea Lane

Area with respect to Sea Lane (in NM^2)	
Ship 1 Track	47.53
Ship 2 Track	57.17
Ship 3 Track	536.54
Ship 4 Track	146.21
Ship 5 Track	2320.04
Ship 6 Track	2184.52

It is clear from Figure 21 that as ship tracks five and six strayed far away from the sea lane, they exhibit a very large area metrics, whereas ships which stayed closer to the sea lane (tracks one through four) exhibit comparatively smaller area metrics. This concept of finding areas between ships' tracks and sea lanes forms the fundamental contribution of this thesis.

C. DEFINING A SEA LANE USING AREA METRIC AND MODIFIED K-MEANS ALGORITHM

We defined sea lanes using historic data of ships' tracks. The procedure for defining the sea lanes was defined in Chapter III, Section B. We start by generating data for ten different ships' tracks in MATLAB. By visual inspection of Figure 22, it is clear that these ships' tracks belong to two different sea lanes. We will try to classify these ships' tracks into two different sea lanes A and B using the algorithm discussed earlier.

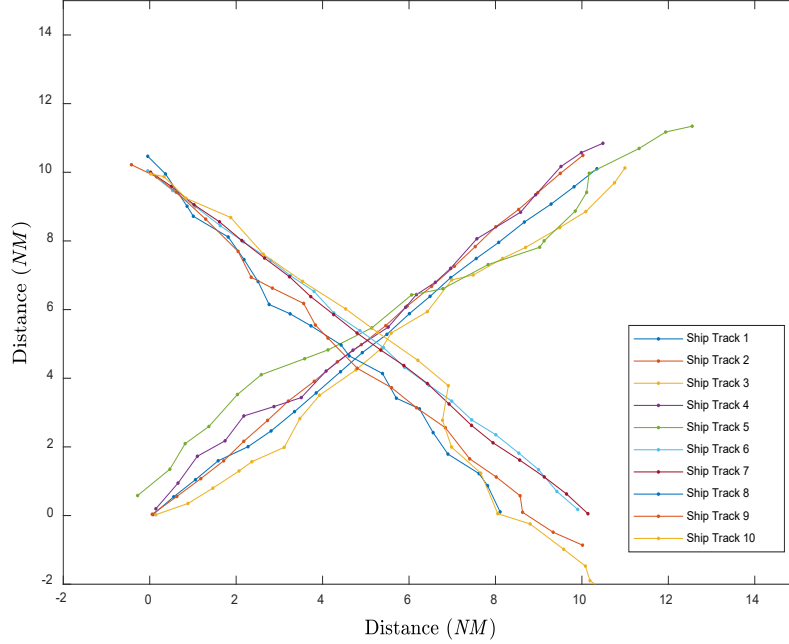


Figure 22. Ships' Tracks in Sea Lanes A and B

We start by calculating the area between each ship's track and all the other ships' tracks, i.e., $d(\mathbf{t}_i, \mathbf{t}_j)$ where $i \in \{1, \dots, 10\}$ and $j \in \{1, \dots, 10\}$, using the area calculation algorithm defined in Chapter III, Section A. As there are ten ship tracks, there are a total of 45 distinct areas between the ships' tracks, as shown in Table 9.

As we are classifying ten ships' tracks into one of two sea lanes (A or B), using the modified k -means algorithm, therefore $k = 2$. We pick two reference ships' tracks at a time and suppose these ships' tracks to be sea lanes A and B . We find the area metrics, $d(\mathbf{t}_i, \mathbf{t}_j)$ where $i \in \{1, 2\}$ and $j \in \{1, \dots, 10\}$, of all the ships' tracks with reference to two selected ships' tracks. All the ships' tracks are then assigned to either sea lane A or B based on the minimum total area metric. To demonstrate this concept, we pick ship tracks one and two shown in Figure 22 and find the area metric of all the ships' tracks with respect to both selected ships' tracks. All the ships' tracks are then assigned to either sea lane A or B , based on the smallest area metric, as shown in Table 10. We also keep a record of the sum of the smallest area metric corresponding to this sea lane assignment.

Table 9. Area Metric between Ships' Tracks (in NM^2)

Area between		Track 1	Track 2	Track 3	Track 4	Track 5	Track 6	Track 7	Track 8	Track 9	Track 10
		$d(t_i, t_1)$	$d(t_i, t_2)$	$d(t_i, t_3)$	$d(t_i, t_4)$	$d(t_i, t_5)$	$d(t_i, t_6)$	$d(t_i, t_7)$	$d(t_i, t_8)$	$d(t_i, t_9)$	$d(t_i, t_{10})$
Track 1	$d(t_1, t_j)$	0	2.00	9.11	4.59	14.05	49.69	49.12	53.58	41.57	45.37
Track 2	$d(t_2, t_j)$	2.00	0	7.17	4.23	12.29	51.17	49.22	55.25	43.51	46.27
Track 3	$d(t_3, t_j)$	9.11	7.17	0	6.77	5.39	50.69	49.32	54.37	43.02	43.64
Track 4	$d(t_4, t_j)$	4.59	4.23	6.77	0	11.80	58.47	54.66	60.88	45.95	46.57
Track 5	$d(t_5, t_j)$	14.05	12.29	5.39	11.80	0	57.15	57.63	57.35	49.70	46.44
Track 6	$d(t_6, t_j)$	49.69	51.17	50.69	58.47	57.15	0	2.00	6.01	2.65	10.93
Track 7	$d(t_7, t_j)$	49.12	49.22	49.32	54.66	57.63	2.00	0	6.38	2.50	12.59
Track 8	$d(t_8, t_j)$	53.58	55.25	54.37	60.88	57.35	6.01	6.38	0	6.96	12.87
Track 9	$d(t_9, t_j)$	41.57	43.51	44.02	45.95	49.70	2.65	2.50	6.96	0	11.44
Track 10	$d(t_{10}, t_j)$	45.37	46.27	43.64	46.57	46.44	10.93	12.59	12.87	11.44	0

Table 10. Sea Lane Assignment Based on Minimum Area Metric

		Track 1	Track 2	Track 3	Track 4	Track 5	Track 6	Track 7	Track 8	Track 9	Track 10
		$d(t_i, t_1)$	$d(t_i, t_2)$	$d(t_i, t_3)$	$d(t_i, t_4)$	$d(t_i, t_5)$	$d(t_i, t_6)$	$d(t_i, t_7)$	$d(t_i, t_8)$	$d(t_i, t_9)$	$d(t_i, t_{10})$
Reference Track 1	$d(t_1, t_j)$	0	2.00	9.11	4.59	14.05	49.69	49.12	53.58	41.57	45.37
Reference Track 2	$d(t_2, t_j)$	2.00	0	7.17	4.23	12.29	51.17	49.22	55.25	43.51	46.27
Sea lane = A if $d(t_1, t_j) < d(t_2, t_j)$ otherwise B		A	B	B	A	B	A	A	A	A	A
Smallest area metric $\min_i(d(t_i, t_j))$		0	0	7.17	4.23	12.29	49.69	49.12	53.58	41.57	45.37
Sum of smallest area metrics $s = \sum_{j=1}^{10} \min_i [d(t_i, t_j)]$						263.02					

Seven ships' tracks are classified into sea lane A , whereas three ships' tracks are classified into sea lane B , as shown in Table 10. The same process is followed for all 45 possible combinations of selecting two reference ships' tracks. We now have 45 different classification models for all ships' tracks, which are shown in Table 11.

From Table 11, we observe that the smallest total area metric value, $s = 47.22$, corresponds to the classification of five ships' tracks in sea lane A and five ships' tracks in sea lane B , which is the optimum classification of these ships' tracks, as is also obvious from Figure 23.

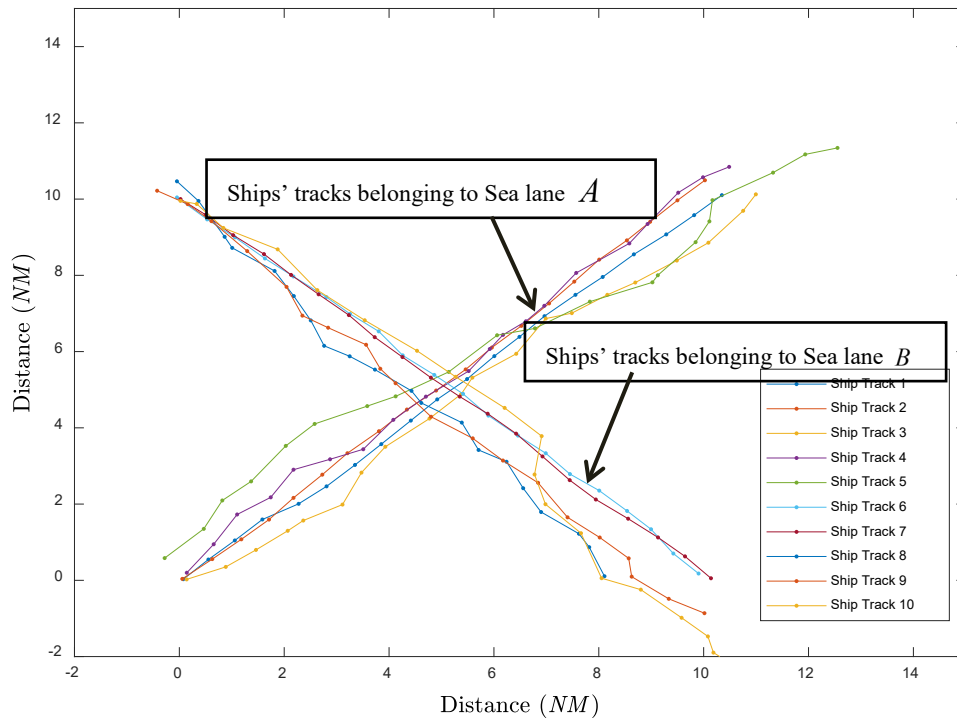


Figure 23. Ships' Tracks Classified into Sea Lane A and B

Table 11. Classification of Ships' Tracks into Sea Lanes Based on Minimum Value of S (Sum of Smallest Area Metrics)

Combination Number	Sea Lanes (Reference Tracks)	Classification of Tracks into Sea Lane A or Sea Lane B										S
		Track 1	Track 2	Track 3	Track 4	Track 5	Track 6	Track 7	Track 8	Track 9	Track 10	
1.	(1,2)	A	B	B	A	B	A	A	A	A	A	263.02
2.	(1,3)	A	A	B	A	B	A	A	A	A	B	248.19
3.	(1, 4)	A	A	B	B	B	A	A	A	A	A	258.45
4.	(1, 5)	A	A	B	A	B	A	A	A	A	A	249.88
5.	(1,6)	A	A	A	A	A	B	B	B	B	B	51.30
6.	(1, 7)	A	A	A	A	A	B	B	B	B	B	53.25
7.	(1, 8)	A	A	A	A	A	B	B	B	B	B	61.96
8.	(1, 9)	A	A	A	A	A	B	B	B	B	B	53.31
9.	(1,10)	A	A	A	A	A	B	B	B	B	B	77.59
10.	(2, 3)	A	A	B	A	B	A	B	A	B	B	259.30
11.	(2, 4)	A	A	B	B	B	A	A	A	B	A	270.92
12.	(2, 5)	A	A	B	A	B	A	A	A	B	B	265.85
13.	(2, 6)	A	A	A	A	A	B	B	B	B	B	47.22
14.	(2, 7)	A	A	A	A	A	B	B	B	B	B	49.18
15.	(2,8)	A	A	A	A	A	B	B	B	B	B	57.88
16.	(2, 9)	A	A	A	A	A	B	B	B	B	B	49.24
17.	(2, 10)	A	A	A	A	A	B	B	B	B	B	73.52
18.	(3, 4)	B	B	A	B	A	A	A	A	B	A	263.31
19.	(3, 5)	A	A	A	A	B	A	A	A	A	A	272.46
20.	(3, 6)	A	A	A	A	A	B	B	B	B	B	50.01
21.	(3, 7)	A	A	A	A	A	B	B	B	B	B	51.96
22.	(3, 8)	A	A	A	A	A	B	B	B	B	B	60.67
23.	(3, 9)	A	A	A	A	A	B	B	B	B	B	52.03
24.	(3, 10)	A	A	A	A	A	B	B	B	B	B	76.31

Combination Number	Sea Lanes (Reference Tracks)	Classification of Tracks into Sea Lane <i>A</i> or Sea Lane <i>B</i>										<i>S</i>
		Track 1	Track 2	Track 3	Track 4	Track 5	Track 6	Track 7	Track 8	Track 9	Track 10	
25.	(4, 5)	<i>A</i>	<i>A</i>	<i>B</i>	<i>A</i>	<i>B</i>	<i>A</i>	<i>A</i>	<i>B</i>	<i>A</i>	<i>B</i>	275.43
26.	(4, 6)	<i>A</i>	<i>A</i>	<i>A</i>	<i>A</i>	<i>A</i>	<i>B</i>	<i>B</i>	<i>B</i>	<i>B</i>	<i>B</i>	48.94
27.	(4, 7)	<i>A</i>	<i>A</i>	<i>A</i>	<i>A</i>	<i>A</i>	<i>B</i>	<i>B</i>	<i>B</i>	<i>B</i>	<i>B</i>	50.90
28.	(4, 8)	<i>A</i>	<i>A</i>	<i>A</i>	<i>A</i>	<i>A</i>	<i>B</i>	<i>B</i>	<i>B</i>	<i>B</i>	<i>B</i>	59.61
29.	(4, 9)	<i>A</i>	<i>A</i>	<i>A</i>	<i>A</i>	<i>A</i>	<i>B</i>	<i>B</i>	<i>B</i>	<i>B</i>	<i>B</i>	50.96
30.	(4, 10)	<i>A</i>	<i>A</i>	<i>A</i>	<i>A</i>	<i>A</i>	<i>B</i>	<i>B</i>	<i>B</i>	<i>B</i>	<i>B</i>	75.24
31.	(5, 6)	<i>A</i>	<i>A</i>	<i>A</i>	<i>A</i>	<i>A</i>	<i>B</i>	<i>B</i>	<i>B</i>	<i>B</i>	<i>B</i>	65.10
32.	(5, 7)	<i>A</i>	<i>A</i>	<i>A</i>	<i>A</i>	<i>A</i>	<i>B</i>	<i>B</i>	<i>B</i>	<i>B</i>	<i>B</i>	67.06
33.	(5, 8)	<i>A</i>	<i>A</i>	<i>A</i>	<i>A</i>	<i>A</i>	<i>B</i>	<i>B</i>	<i>B</i>	<i>B</i>	<i>B</i>	75.77
34.	(5, 9)	<i>A</i>	<i>A</i>	<i>A</i>	<i>A</i>	<i>A</i>	<i>B</i>	<i>B</i>	<i>B</i>	<i>B</i>	<i>B</i>	67.12
35.	(5, 10)	<i>A</i>	<i>A</i>	<i>A</i>	<i>A</i>	<i>A</i>	<i>B</i>	<i>B</i>	<i>B</i>	<i>B</i>	<i>B</i>	91.40
36.	(6, 7)	<i>B</i>	<i>B</i>	<i>B</i>	<i>B</i>	<i>A</i>	<i>A</i>	<i>B</i>	<i>A</i>	<i>B</i>	<i>A</i>	278.92
37.	(6, 8)	<i>A</i>	<i>A</i>	<i>A</i>	<i>A</i>	<i>A</i>	<i>A</i>	<i>A</i>	<i>B</i>	<i>A</i>	<i>A</i>	282.74
38.	(6, 9)	<i>B</i>	<i>B</i>	<i>B</i>	<i>B</i>	<i>B</i>	<i>A</i>	<i>A</i>	<i>A</i>	<i>B</i>	<i>A</i>	242.68
39.	(6, 10)	<i>B</i>	<i>B</i>	<i>B</i>	<i>B</i>	<i>B</i>	<i>A</i>	<i>A</i>	<i>A</i>	<i>A</i>	<i>B</i>	238.95
40.	(7, 8)	<i>A</i>	<i>A</i>	<i>A</i>	<i>A</i>	<i>B</i>	<i>A</i>	<i>A</i>	<i>B</i>	<i>A</i>	<i>A</i>	276.81
41.	(7, 9)	<i>B</i>	<i>B</i>	<i>B</i>	<i>B</i>	<i>B</i>	<i>A</i>	<i>A</i>	<i>A</i>	2	<i>B</i>	243.60
42.	(7, 10)	<i>B</i>	<i>B</i>	<i>B</i>	<i>B</i>	<i>B</i>	<i>A</i>	<i>A</i>	<i>A</i>	<i>A</i>	<i>B</i>	239.20
43.	(8, 9)	<i>B</i>	<i>B</i>	<i>B</i>	<i>B</i>	<i>B</i>	<i>B</i>	<i>B</i>	<i>A</i>	<i>B</i>	<i>B</i>	240.38
44.	(8, 10)	<i>B</i>	<i>B</i>	<i>B</i>	<i>B</i>	<i>B</i>	<i>A</i>	<i>A</i>	<i>A</i>	<i>A</i>	<i>B</i>	247.66
45.	(9, 10)	<i>A</i>	<i>A</i>	<i>A</i>	<i>A</i>	<i>B</i>	<i>A</i>	<i>A</i>	<i>A</i>	<i>A</i>	<i>B</i>	232.63

Based on the classification of these ten ships' tracks into sea lanes, we take the average of all ships' tracks corresponding to one particular sea lane and find the data points or track for that sea lane. The two sea lanes found from this algorithm are shown in Figure 24.

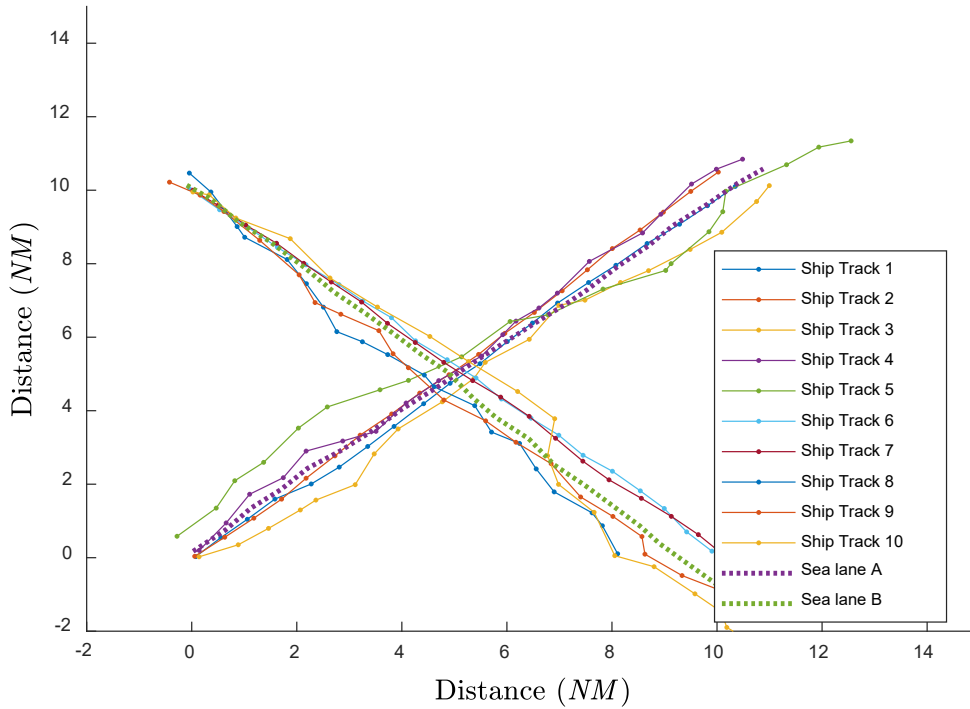


Figure 24. Sea Lane *A* and Sea Lane *B* formed by Averaging Tracks Belonging to these Sea Lanes

D. CLASSIFICATION OF SHIPS' TRACKS INTO SEA LANES USING AREA METRIC ALGORITHM

For this section, we assume that we have already found the sea lanes in which the ships' tracks are to be classified. In Chapter III, Section C, we described three different ways of classifying ships' tracks into sea lanes. Results from these algorithms are as follows.

1. User-Defined Threshold

Using this algorithm, ships' tracks are classified into different sea lanes based on a user-defined threshold for the area metric with respect to each sea lane. We defined two sea lanes (sea lanes A and B) and ten different ships' tracks, as shown in Figure 25.

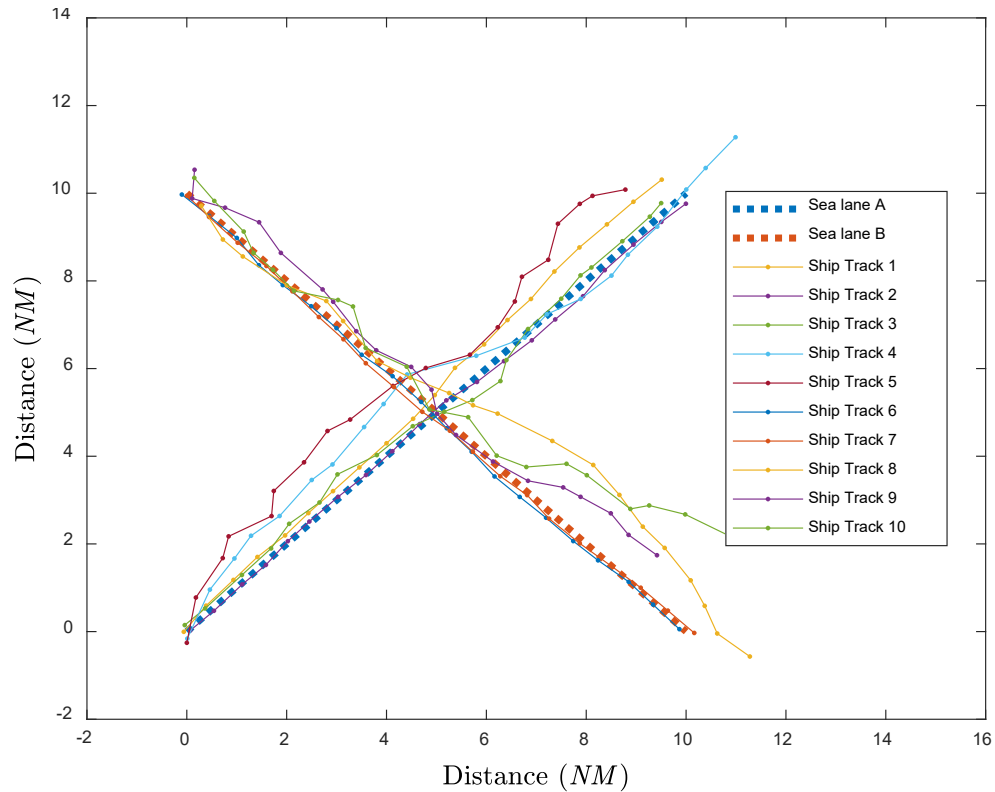


Figure 25. Sea Lanes A and B with Ships' Tracks

First we find the area metric of all the ships' tracks with respect to each of the sea lanes using the area metric algorithm. For each sea lane, we defined an area threshold value. If the area of any of the ten ships' tracks is less than the threshold value, the ships' track is classified as being in that sea lane. For sea lane A , we described the threshold value $Th_A = 20$, and for sea lane B , we defined the threshold value $Th_B = 15$. The threshold values could be different based on the region in maritime domain, e.g., in territorial waters we may

choose a smaller value for the threshold compared to the threshold value for open ocean as vessels are spaced closer in territorial waters than open ocean. The result of the classification of ten ships tracks into two sea lanes using this technique is shown in Table 12.

Table 12. Classification of Ships' Tracks into Sea Lanes Using a User-Defined Threshold

	Area of ships' tracks with Sea Lane A $d(\mathbf{s}_A, \mathbf{t}_j)$ (in NM^2)	Area of ships' tracks with Sea Lane B $d(\mathbf{s}_B, \mathbf{t}_j)$ (in NM^2)	Classification A if $d(\mathbf{s}_A, \mathbf{t}_j) \leq Th_A=20$ B if $d(\mathbf{s}_B, \mathbf{t}_j) \leq Th_B=15$
Ship Track 1	1.04	48.75	A
Ship Track 2	3.29	47.76	A
Ship Track 3	3.89	59.66	A
Ship Track 4	2.65	47.90	A
Ship Track 5	13.77	49.73	A
Ship Track 6	49.24	0.64	B
Ship Track 7	50.05	0.68	B
Ship Track 8	46.26	1.93	B
Ship Track 9	52.87	16	None
Ship Track 10	52.13	12.92	B

As this technique is dependent on the user-defined threshold for the area, if the area of a ship's track with respect to each sea lane is greater than the threshold value of each sea lane, the ship's track are not classified into any of the sea lanes. In Table 12, the areas of ship track nine with respect to sea lane A and sea lane B are greater than the respective threshold values of each sea lane; therefore, ship track nine is not classified into either of the sea lanes.

2. The Modified k -means Clustering Algorithm

The algorithm for the classification of ships' tracks by sea lane using the modified k -means algorithm is defined in Chapter III, Section C. In this section, we classify ten ships' tracks

into one of two sea lanes using this algorithm. This algorithm classifies ships' tracks into sea lanes yielding a minimum area metric. Ships' tracks to be classified into sea lanes are shown in Figure 26, and their respective areas with respect to sea lane A and sea lane B are given in Table 13.

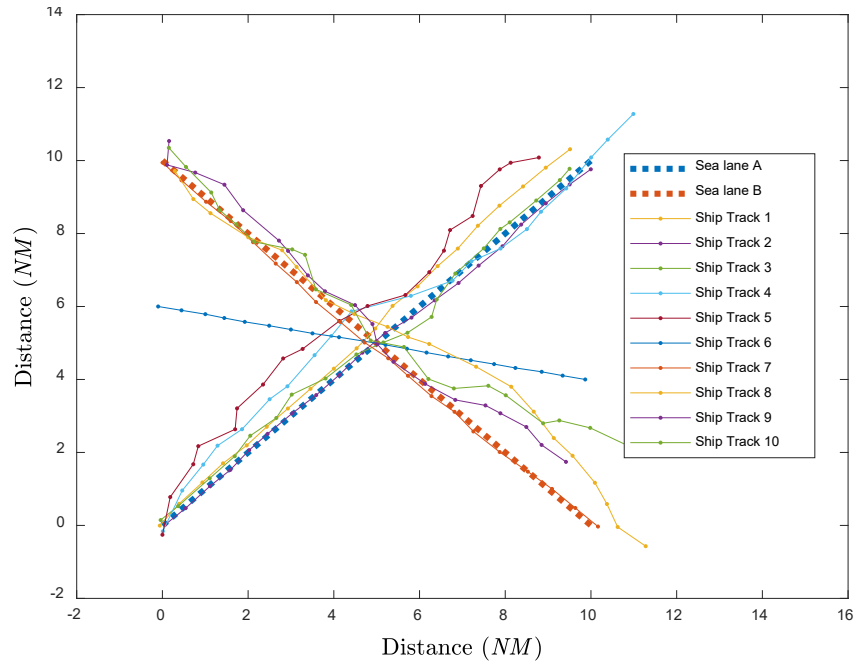


Figure 26. Classifying Ships' Tracks into Sea Lanes Using Modified k -means Algorithm

It is clear from Figure 26 and Table 13 that the ships' tracks closer to sea lane A have been classified as belonging to this sea lane, whereas ships' tracks closer to sea lane B have been classified as belonging to sea lane B . This is also valid for ship track six, which has smaller area metric with respect to sea lane B as compared to sea lane A and is, therefore, classified in sea lane B .

Table 13. Ships' Classification into Sea Lanes Using Modified k -means Algorithm

	Area of ships' tracks with Sea Lane A $d(\mathbf{s}_A, \mathbf{t}_j)$ (in NM^2)	Area of ships' tracks with Sea Lane B $d(\mathbf{s}_B, \mathbf{t}_j)$ (in NM^2)	Classification = A if $d(\mathbf{s}_A, \mathbf{t}_j) \leq d(\mathbf{s}_B, \mathbf{t}_j)$ Otherwise B
Ship Track 1	0.68	49.26	A
Ship Track 2	0.82	51.52	A
Ship Track 3	7.25	55.29	A
Ship Track 4	2.43	53.76	A
Ship Track 5	9.17	37.22	A
Ship Track 6	29.89	19.88	B
Ship Track 7	49.72	2.77	B
Ship Track 8	53.42	12.3	B
Ship Track 9	47.96	3.87	B
Ship Track 10	54.84	3.75	B

3. Bayes' Theorem

This technique uses Bayes' Theorem of conditional probability to classify ships' tracks into sea lanes. For this section, we again assume that we are classifying ships' tracks into one of two sea lanes. To explain this concept we present two cases using different training sets.

a. Case 1 – Equal Standard Deviations for Both Sea Lanes.

We start with a simple case in which the nine ships' tracks represented in Figure 27 are to be classified into sea lane A or sea lane B . The area metrics of all ships' tracks with reference to both sea lanes are shown in Table 14.

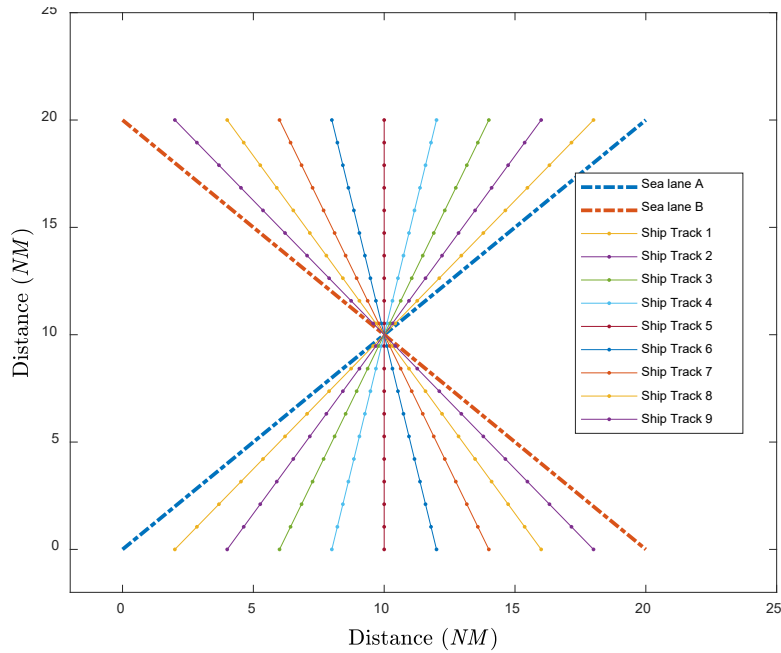


Figure 27. Ships' Tracks and Two Sea Lanes

Table 14. Area of Ships' Tracks with Respect to Sea Lanes A and B

	Area of Ships' Tracks with Sea Lane A $d(s_A, t_j)$ (in NM^2)	Area of Ships' Tracks with Sea Lane B $d(s_B, t_j)$ (in NM^2)
Ship Track 1	20	180
Ship Track 2	40	160
Ship Track 3	60	140
Ship Track 4	80	120
Ship Track 5	100	100
Ship Track 6	120	80
Ship Track 7	140	60
Ship Track 8	160	40
Ship Track 9	180	20

As explained in Chapter III, Section C, we estimate the conditional PDFs of the area metric for ships' tracks from sea lane A $f_{X,Y|A}(x,y)$ and the conditional PDF of the area metric for ships' tracks from sea lane B $f_{X,Y|B}(x,y)$ using (22) and (23). The parameters for computing these PDFs (e.g., means and standard deviations) are calculated from historic data of these sea lanes. For this section, we assume the values of these parameters, which are shown in Table 15.

Table 15. Parameters of Conditional PDFs

Parameters of PDFs	Value
Mean for area metric of ships' track with respect to sea lane A given that track belongs to sea lane A - $\mu_{X A}$	0
Mean for area metric of ships' track with respect to sea lane A given that track belongs to sea lane B - $\mu_{X B}$	30
Mean for area metric of ships' track with respect to sea lane B given that track belongs to sea lane A - $\mu_{Y A}$	30
Mean for area metric of ships' track with respect to sea lane B given that track belongs to sea lane B - $\mu_{Y B}$	0
Standard deviation for area metric of ships' track with respect to sea lane A given that track belongs to Sea lane A - $\sigma_{X A}$	50
Standard deviation for area metric of ships' track with respect to sea lane A given that track belongs to sea lane B $\sigma_{X B}$	50
Standard deviation for area metric of ships' track with respect to sea lane B given that track belongs to sea lane A - $\sigma_{Y A}$	50
Standard deviation for area metric of ships' track with respect to sea lane B given that track belongs to sea lane B - $\sigma_{Y B}$	50

Notice that all the conditional standard deviations are same, i.e., 50 for this case. We also assume that both the sea lanes are equally likely (i.e., both have equal prior probabilities $P(A) = P(B) = 0.5$). We find the probability of ships' tracks being in sea lane A given the area metrics, i.e., $P(A | X = x, Y = y)$, and the probability of ships' tracks being in sea lane B given the area metrics, i.e., $P(B | X = x, Y = y)$, using (20) and (21). Results for these probabilities are shown in Table 16. It is clear from Figure 27 that ship track one

is very close to sea lane A and, therefore, has a smaller area metric with respect to sea lane A , whereas it has largest area with respect to sea lane B . Consequently, the probability of ship track one being in sea lane A is high (i.e., 0.87) as compared to its probability of being in sea lane B (i.e., 0.13). Similarly, as ships' tracks one through nine move away from sea lane A and closer to sea lane B , their probability of being in sea lane A also decreases, whereas the probability of their being in sea lane B increases. Also notice that in the case of ship track five where its area metric with respect to both the sea lanes is equal, its probability of being in both the sea lanes is equal, i.e., 0.5. As discussed earlier in Chapter III, Section C, the sea lanes A and B are exhaustive and mutually exclusive, which is clear from Table 16.

Table 16. Probabilities of Sea Lane A and Sea Lane B Given Ships' Track Areas

	Area of Ships' Tracks with Sea Lane A $x = d(\mathbf{s}_A, \mathbf{t}_j)$ (in NM^2)	Area of Ships' Tracks with Sea Lane B $y = d(\mathbf{s}_B, \mathbf{t}_j)$ (in NM^2)	Probability of Ships' Tracks being in Sea Lane A $P(A X = x, Y = y)$	Probability of Ships' Tracks being in Sea Lane B $P(B X = x, Y = y)$
Ship Track 1	20	180	0.87	0.13
Ship Track 2	40	160	0.81	0.19
Ship Track 3	60	140	0.72	0.28
Ship Track 4	80	120	0.61	0.39
Ship Track 5	100	100	0.50	0.50
Ship Track 6	120	80	0.39	0.61
Ship Track 7	140	60	0.28	0.72
Ship Track 8	160	40	0.19	0.81

From (22) and (23), it is obvious that the conditional PDFs $f_{X,Y|A}(x,y)$ and $f_{X,Y|B}(x,y)$ depend on the parameters (means and standard deviations) of ships' tracks from training data. In this case, to explain the concept clearly, we chose ship's track area metric means ($\mu_{X|A}$ and $\mu_{X|B}$) as well as their standard deviation ($\sigma_{X|A}, \sigma_{X|B}, \sigma_{Y|A},$ and $\sigma_{Y|B}$) as being equal.

b. Case 2 - Different Standard Deviations for Both Sea Lanes

For this case, we calculated the parameters of conditional PDFs used in (22) and (23), i.e., ships' tracks area metric means ($\mu_{X|A}$ and $\mu_{X|B}$) as well as their standard deviations ($\sigma_{X|A}, \sigma_{X|B}, \sigma_{Y|A},$ and $\sigma_{Y|B}$) from training data of ships' tracks in two sea lanes as shown in Figure 28.

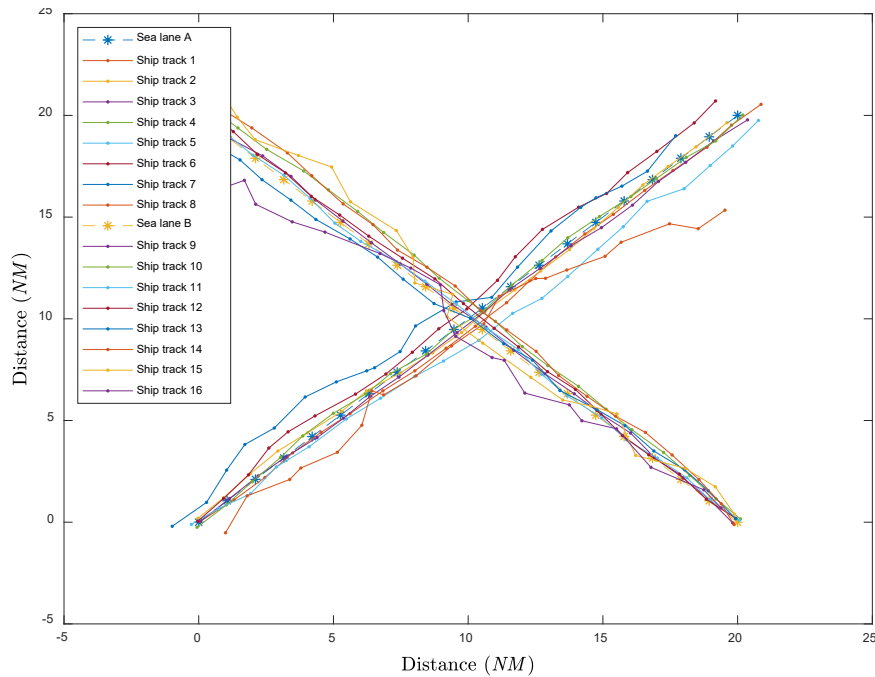


Figure 28. Training Data of Ships' Tracks in Sea lanes A or B

The parameters to compute conditional PDFs of ships' tracks are recorded in Table 17. After obtaining the conditional PDFs' parameters, we compute the probability of the nine ships' tracks belonging to the test set, as shown in Figure 29. The probabilities of these ships' tracks being in sea lane A or sea lane B are computed using (19) and (20) and are shown in Table 18.

Table 17. Parameters of Conditional PDFs Computed from Training Data

Parameters of PDFs	Value
Mean for area metric of ships' track with respect to sea lane A given that track belongs to sea lane A - $\mu_{X A}$	0
Mean for area metric of ships' track with respect to sea lane A given that track belongs to sea lane B - $\mu_{X B}$	192.99
Mean for area metric of ships' track with respect to sea lane B given that track belongs to sea lane A - $\mu_{Y A}$	197.01
Mean for area metric of ships' track with respect to sea lane B given that track belongs to sea lane B - $\mu_{Y B}$	0
Standard deviation for area metric of ships' track with respect to sea lane A given that track belongs to sea lane A - $\sigma_{X A}$	12.76
Standard deviation for area metric of ships' track with respect to sea lane A given that track belongs to sea lane B - $\sigma_{X B}$	6.31
Standard deviation for area metric of ships' track with respect to sea lane B given that track belongs to sea lane A - $\sigma_{Y A}$	5.96
Standard deviation for area metric of ships' track with respect to sea lane B given that track belongs to sea lane B - $\sigma_{Y B}$	26.22

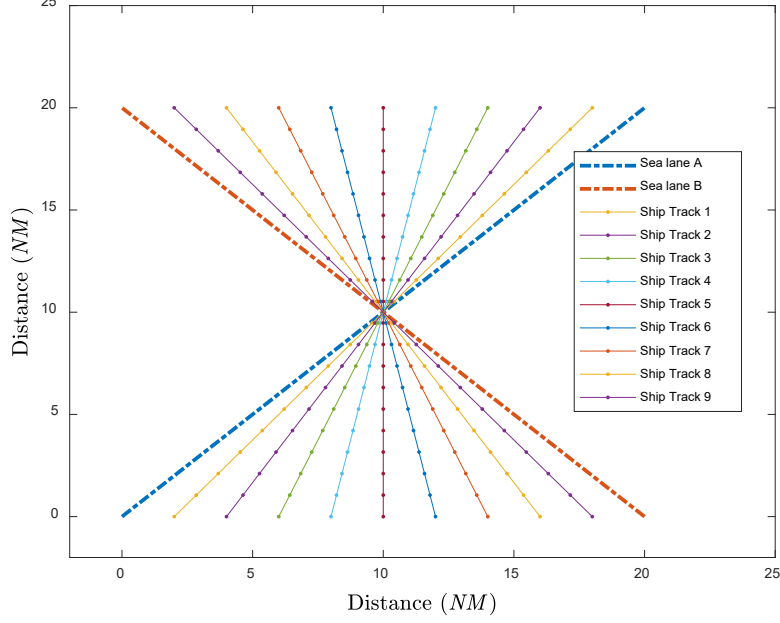


Figure 29. Test Data Set of Ships' Tracks and Sea Lanes A and B

Notice that even though the area metric of the nine ships' tracks with respect to sea lanes A and B in case 1 and case 2 are exactly the same, they have different probabilities of being in sea lanes A and B , as shown in Tables 16 and 18. If we compare ship track five in both the cases, its area metrics with respect to sea lanes A and B are the same, as shown in Tables 16 and 18. Yet, it has equal probabilities of being in both sea lanes (i.e., 0.50 in case 1), because it is two standard deviations away from both sea lane A and B . In case 2, however, the probability of ship track five being in sea lane A is very small and is, therefore, classified as belonging to sea lane B . Using (22) to compute the conditional PDF of the area metric of ship track five given sea lane A , i.e. $f_{X,Y|A}(x,y)$, we have $(|x - \mu_{x|A}| / \sigma_{x|A}) = (|100 - 0| / 12.76) = 7.83$, which implies that X is 7.83 standard deviations away from mean $\mu_{x|A}$, and $(|y - \mu_{y|A}| / \sigma_{y|A}) = (|100 - 197.01| / 5.96) = 16.2$ implies that Y is 16.2 standard deviations away from mean $\mu_{y|A}$. Using these numbers in (22), we get a factor of $e^{-8.99}$, which is a very small number. Similarly, the conditional PDF of the area metric for ship track five given sea lane B , i.e. $f_{X,Y|B}(x,y)$ computed using (23), has a

factor of $e^{-7.61}$, which is almost 3.97 times greater than $e^{-8.99}$. Using (20) and (21), we classify ship track five as belonging to sea lane B with a probability of 0.99.

Table 18. Posterior Probabilities of Sea Lane A and Sea Lane B Given Ships' Tracks Areas

	Area of Ships' Tracks with Sea Lane A $x = d(\mathbf{s}_A, \mathbf{t}_j)$ (in NM^2)	Area of Ships' Tracks with Sea Lane B $y = d(\mathbf{s}_B, \mathbf{t}_j)$ (in NM^2)	Probability of Ships' Tracks being in Sea Lane A $P(A X = x, Y = y)$	Probability of Ships' Tracks being in Sea Lane B $P(B X = x, Y = y)$
Ship Track 1	20	180	0.99	3.84e-172
Ship Track 2	40	160	0.99	3.13e-126
Ship Track 3	60	140	0.99	5.56e-79
Ship Track 4	80	120	0.99	2.13e-30
Ship Track 5	100	100	5.48e-21	0.99
Ship Track 6	120	80	3.08e-72	0.99
Ship Track 7	140	60	7.83e-125	0.99
Ship Track 8	160	40	9.16e-179	0.99
Ship Track 9	180	20	4.94e-234	0.99

From the scenarios presented in Sections A and B in this chapter, we conclude that ships' tracks are classified into sea lanes nicely (with high and low probabilities) when the standard deviations are small. Nevertheless, since we started with the assumption that sea lanes A and B are exhaustive and mutually exclusive, ships' tracks will be forcefully classified into one of the sea lanes when standard deviations for the sea lanes are smaller, even though the ships' tracks may not belong to any of the sea lanes.

E. ANOMALY DETECTION

We are also interested in analyzing two types of anomalies. The algorithms for detecting these anomalies are discussed in Chapter III, Section D. These two types are analyzed, and the results are discussed in the following sections.

1. Detecting Outliers in Standard Sea Lanes Using Area Metric and Bayes' Theorem

The algorithm for detecting outliers in standard sea lanes is discussed in Chapter III, Section C, where we classified ships' tracks into sea lanes based on their area metric by calculating their conditional probabilities using (20) and (21). For this section, we again consider two sea lanes (A and B) in a particular maritime area. We calculate the parameters of conditional PDFs of both the sea lanes from historical training data. For this section we assume the same historical ships' tracks data shown in Figure 28 and the parameters of the conditional PDFs (means and standard deviation) shown in Table 17. We apply the parameters obtained from the training data to the test set of ships' tracks to find their probabilities of being in sea lanes A and B . The area metrics of the test set ships' tracks with respect to both sea lanes and their posterior conditional probabilities of being in each sea lane are shown in Table 19.

From Tables 18 and 19, it is evident that most of the ships' tracks have a high probability of being in one of the sea lanes and a low probability of being in the other. In Table 19, however, ships' tracks five and six have a relatively low probability of being in either sea lane. This may be considered an outlier when compared to the historical training data.

Table 19. Anomaly Detection Based on Posterior Probabilities of Sea Lane A and Sea Lane B Given Ships' Tracks Areas

	Area of Ships' Tracks with Sea Lane A $x = d(\mathbf{s}_A, \mathbf{t}_j)$ (in NM^2)	Area of Ships' Tracks with Sea Lane B $y = d(\mathbf{s}_B, \mathbf{t}_j)$ (in NM^2)	Probability of Ships' Tracks being in Sea Lane A $P(A X = x, Y = y)$	Probability of Ships' Tracks being in Sea Lane B $P(B X = x, Y = y)$
Ship Track 1	97	404	3.56e-20	1
Ship Track 2	104	397	2.09e-06	0.99
Ship Track 3	374	127	1.10e-08	0.99
Ship Track 4	80	120	0.99	2.13e-30
Ship Track 5	107	394	0.51	0.49
Ship Track 6	370	131	0.32	0.68
Ship Track 7	369	132	0.97	0.03

2. Detecting Ships Following One Another Using Area Metric Algorithm and User-Defined Threshold

The algorithm for detecting ships following one another in a sea lane is explained in Chapter III, Section D. The area metric between ships' tracks in this algorithm accounts for their spatial as well as temporal locality. For example, we first consider two ships' tracks that are exactly on the same sea lane and, therefore, have a zero area metric in the spatial domain as shown in Figure 30. If we account for only the spatial domain of both these ships' tracks, the area metric between these ships' tracks is obviously zero; however, if we consider both of these ships' tracks being in this sea lane at different times and account for their temporal separation in the area metric calculation, it yields a non-zero area metric between them as shown in Figure 31.

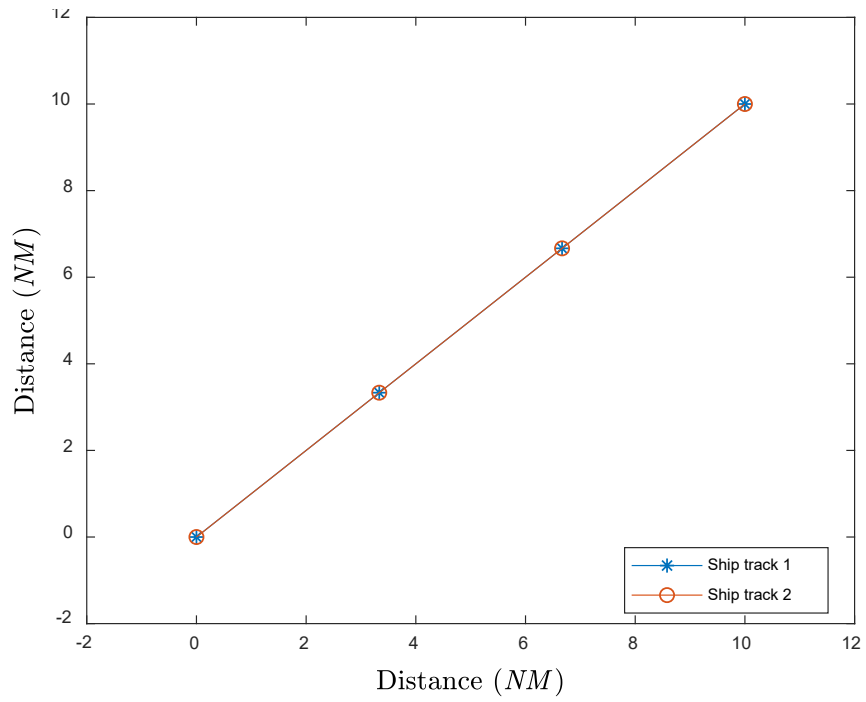


Figure 30. Ships' Tracks in the Same Sea Lane in Spatial Domain

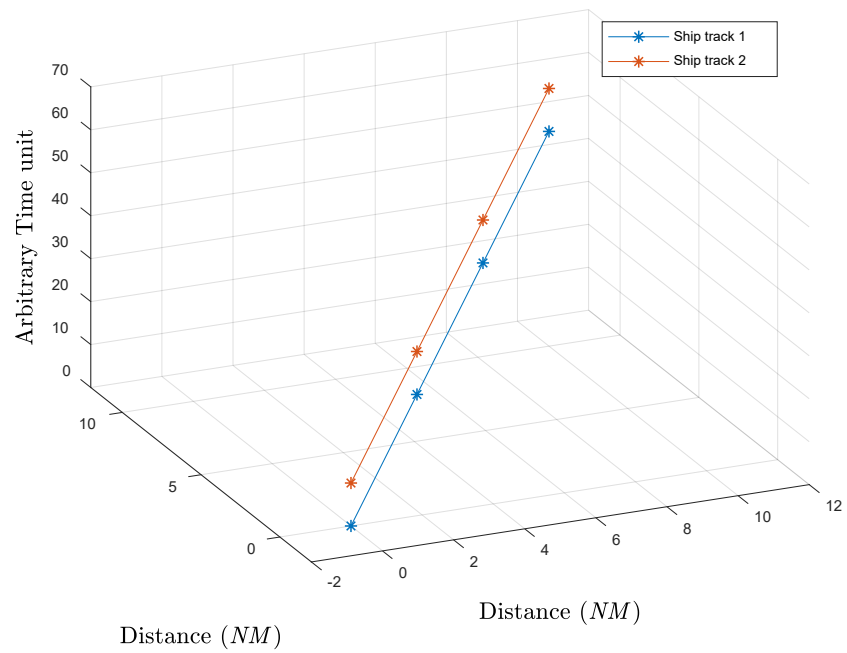


Figure 31. Ships' Tracks in the Same Sea Lane at Different Times

To address this scenario, we set a threshold X_{Th} below which all the area metric values between ships' tracks correspond to ships following each other. The user-defined threshold X_{Th} may be considered the acceptance level of ships following one another. It can be seen in Figure 32 that ship tracks one and two stay very close to one another; therefore, the area metric between both of these ships' tracks is very small as shown in Table 20.

In this case, if we select the threshold value of $X_{Th}=5$, we are able to detect ship tracks one and two following each other based on their area metric $d(\mathbf{t}_1, \mathbf{t}_2)$. Similarly, it can be seen from Figure 32 that ship track one and five strayed away from each other both in spatial as well as temporal domain. From Table 20, we see that they exhibit a large value of area metric, i.e., $d(\mathbf{t}_1, \mathbf{t}_5) = 22.04$, which is greater than X_{Th} and, therefore, are not classified as exhibiting following or paralleling behavior.

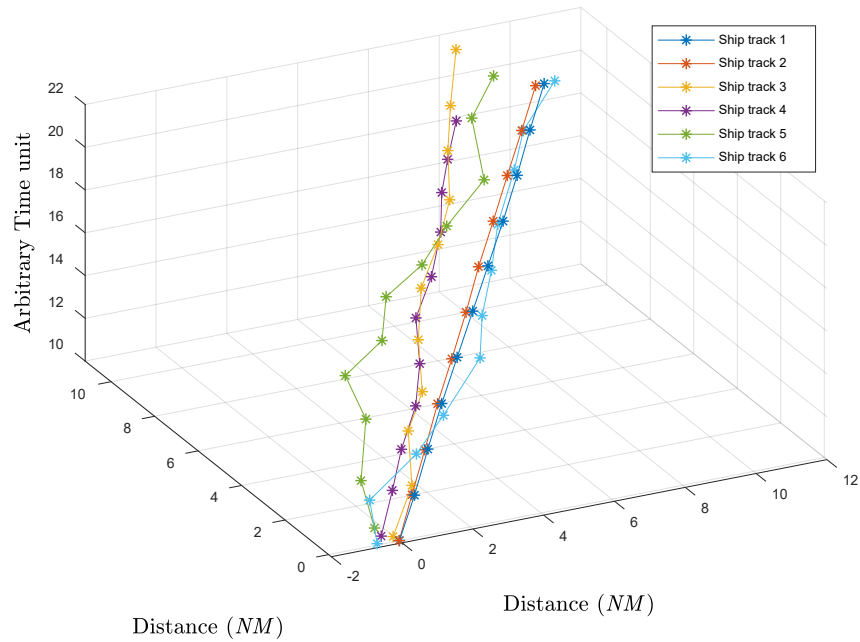


Figure 32. Anomaly Detection of Ships Following One Another

Table 20. Area Metric between Ships' Tracks (in NM^2)

Area b/w		Track 1	Track 2	Track 3	Track 4	Track 5	Track 6
		$d(t_i, t_1)$	$d(t_i, t_2)$	$d(t_i, t_3)$	$d(t_i, t_4)$	$d(t_i, t_5)$	$d(t_i, t_6)$
Track 1	$d(t_1, t_j)$	0	1.15	14.91	15.70	22.04	11.73
Track 2	$d(t_2, t_j)$	1.15	0	14.22	15.95	21.38	11.22
Track 3	$d(t_3, t_j)$	14.91	14.22	0	29.3	9.96	18.46
Track 4	$d(t_4, t_j)$	15.70	15.95	29.37	0	35.74	18.24
Track 5	$d(t_5, t_j)$	22.04	21.38	9.96	35.74	0	22.76
Track 6	$d(t_6, t_j)$	11.73	11.22	18.46	18.24	22.76	0

V. CONCLUSION

The prime objective of this thesis was to develop an algorithm to improve MDA by autonomously analyzing vessels' behavior and detection of outliers. Synthetic data was used in this thesis instead of AIS data to avoid data inconsistencies. Unlike most of the work done in this research area, in this thesis research we did not process the actual ships' tracks data for classification or anomaly detection but rather transformed the ships' tracks data into an area metric between ships' tracks.

These area metric values are then used to identify sea lanes using a modified k -means algorithm. These sea lanes are found by averaging all the ships' tracks belonging to a particular sea lane. Once the sea lanes are identified, area metrics for all the ships' tracks are calculated with respect to the sea lanes, and these ships' tracks are classified by sea lanes based on three different algorithms, i.e., using user-defined thresholds for area metric values in each sea lane, Bayes' theorem, and a modified k -means clustering algorithm.

Two types of anomalies were analyzed in this thesis. Ships not following standard sea lanes were identified by calculating the area metric between ships' tracks and applying Bayes' theorem to calculate their posterior probabilities of being in these sea lanes. Ships not exhibiting high probability of being in any one sea lane were identified as anomalies. A group of ships following one another or paralleling for a significant period of time are considered anomalous and are detected in this thesis using the enhanced area metric algorithm and user-defined thresholds.

A. SIGNIFICANT CONTRIBUTIONS

The work done in this thesis provides three contributions to improve MDA. First, to measure the degree of similarity or distinction between two ships' tracks, an area metric was developed to calculate the area defined by two ships' tracks with a simple method of closing the area on the ends. The development of this area metric is the most significant contribution and is fundamental to this work. This area metric may be a major contribution to the MDA field because it is not sensitive to the ships' tracks position data inconsistencies

(e.g., some ships transmit position reports more frequently than others), and applying conventional machine learning algorithms directly to these positional reports data does not always give useful results. Second, sea lanes were identified in a particular maritime area using area metrics of ships' tracks and applying a modified k -means algorithm to them. The modified k -means clustering algorithm was inspired by the conventional k -means clustering algorithm. The difference is that instead of Euclidian distances, area metrics of ships' tracks were used for grouping.

Finally, an algorithm to calculate the probability of ships being in different sea lanes was developed. This algorithm uses Bayes' theorem of conditional probabilities and the area metric between ships' tracks. Anomalies including ships not following standard sea lanes and groups of ships exhibiting following or paralleling behavior were identified using the area metric between ships' tracks.

B. FUTURE WORK

In this thesis, we focused on the development of algorithms to measure the degree of similarity or distinction between ships' tracks by calculating the area between them. For this purpose, we used synthetic data instead of actual AIS data to avoid data inconsistencies. An extension of this work would be to use actual AIS data for all the algorithms developed herein.

All the algorithms developed in this thesis may work for multiple sea lanes in a particular maritime area. To present the concept clearly, however, we considered two sea lanes in a particular maritime area at a time. In future work, the algorithms developed in this thesis should be expanded to account for many sea lanes in a particular maritime area.

Sea lanes were defined by averaging the ships' tracks classified by sea lanes using the modified k -means clustering algorithm. In this thesis we assumed the same number of positional reports for all ships' tracks; therefore, the process of averaging the ships' tracks belonging to a particular sea lane was simple. By contrast, when actual AIS data is used, the number of positional reports for ships' tracks may be different. In future work, a method

of averaging ships' tracks with different number of positional reports belonging to a particular sea lane needs to be devised.

As noted earlier, the modified k -means clustering algorithm used in this thesis to identify sea lanes was inspired by the conventional k -means clustering algorithm. Use of the conventional algorithm generally assumes we have the prior information about the number of clusters in which the observations are to be classified. In this work, we also assumed that we have prior information about the number of sea lanes in a particular maritime area. An extension of this work would be to find the optimum number of sea lanes in a particular maritime area.

A method of classifying ships' tracks into sea lanes using Bayes' theorem was developed in this thesis. We assumed the distribution of area metrics of ships' tracks with respect to sea lanes, i.e., X and Y , as a Gaussian distribution. Since the area metrics are always non-negative, however, the assumption of being a Gaussian distribution is not true in a strict sense. In future work, instead of a Gaussian distribution, the random variables X and Y may also be modeled with other distribution types such as the lognormal or Poisson distribution.

Anomaly detection of ships' following one another was identified using a user-defined threshold for enhanced area metrics of ships' tracks. We used a sliding window technique for a specific number of past data points for calculating area metrics for all ships' tracks. Nevertheless, the sliding window technique may not work well while using actual AIS data since different ships transmit position reports with different frequencies (e.g., once every five minutes, once every minute, etc.). Thus, an extension of this work would be to explore other techniques of considering specific length (in nautical miles or time) for all ships' tracks.

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX

This appendix contains the MATLAB code for implementing all the algorithms in this thesis.

```
Function triangleArea = triA(twoPts, thirdPt, track,
sealane)

% This function calculates triangular area defined by three
points
% Function input: Three points, Ships' track matrix, Sea
lane matrix
% Fucntion output: Area enclosed by three selected points
% Created by: Tauseef Ashraf, LT CDR PN, September 2018
    Pt1 = track(:, cell2mat(twoPts(1,2))); %Candidate 1
    Pt2 = sealane(:, cell2mat(twoPts(2,2)));%Candidate 2
    if strcmp('track', thirdPt(1))
        Pt3 = track(:, cell2mat(thirdPt(2)));
    elseif strcmp('sealane', thirdPt(1))
        Pt3 = sealane(:, cell2mat(thirdPt(2)));
    else Pt3=thirdPt;
    end

    d12 = euDist(Pt1, Pt2); %Euclidian Distance between
Point 1 and Point 2
    d13 = euDist(Pt1, Pt3); %Euclidian Distance between
Point 1 and Point 3
    d23 = euDist(Pt3, Pt2); %Euclidian Distance between
Point 3 and Point 2

    %Formula for Triangular Area
    s = 0.5*(d12 + d13 + d23);
    triangleArea = sqrt( s*(s-d12)*(s-d13)*(s-d23) );

end
```



```

function [ tkcomset1_x, tkcomset1_y ] = makeTracks(
stan_dev, sealane_fixes, SIZEOFDATA )
% This function is used to simulate ships' tracks by adding
noise to the Sea lane data points
% Function input: Standard deviation for ship's track, Sea
lane
% matrix, Size of data points
% Fucntion output: x abd y coordinates of Ships' tracks
% Created by: Tauseef Ashraf, LT CDR PN, September 2018
for index=1: length(stan_dev)

noisefl1_x(index,:)=conv(stan_dev(index)*randn(1,SIZEOFDATA
),ones(1,200)); %gaussian noise filtered for adding to x-
axis of tracks

tkcomset1_x(index,:)=sealane_fixes(1,:)+(noisefl1_x(index,1
:SIZEOFDATA)); % x-axis of set of ships tracks

noisefl1_y(index,:)=conv(stan_dev(index)*randn(1,SIZEOFDATA
),ones(1,200)); %gaussian noise filtered for adding to x-
axis of tracks

tkcomset1_y(index,:)=sealane_fixes(2,:)+(noisefl1_y(index,1
:SIZEOFDATA)); % y-axis of set of ships tracks
end

```

```
function dist = euDist(X1, X2)

% This function calculates the euclidian distance between
two points
% Function input: Two data points of Sea lane or Ship's
track matrix
% Fucntion output: Euclidian distance between both data
points
% Created by: Tauseef Ashraf, LT CDR PN, September 2018

    diff = abs(X1-X2);
    distSqr = diff'*diff;
    dist = sqrt(distSqr);

end
```

```

function [thirdPt,crossOver] = doTheyCross(twoPts, maybes,
track, sealane, meanstart, trackCntr, sealaneCntr)
% This functions checks for cross over point and calculated
the cross over point between
% ships tracks and sea lane data points
% Function input: Two data points already determined for
new triangular
% area, likely next candidates for third data points,
Ship's track
% matrix, Sea lane matrix, Mean of first data point of
Ship's track and Sea
% lane, Sea lane data point counter, Ship's track data
point counter
% Fucntion output: Crossover if there is a cross over,
Crossover flag
% Created by: Tauseef Ashraf, LT CDR PN, September 2018
[~,sealanenumpts]=size(sealane);
[~,tracknumpts]=size(track);
atEnd=(sealaneCntr >= sealanenumpts) | (trackCntr >=
tracknumpts);
    if ~atEnd

        Pts = [track(:,trackCntr) track(:,trackCntr+1) ...
                sealane(:,sealaneCntr) sealane(:,sealaneCntr+1)];

        mt = (Pts(2,2)-Pts(2,1))/(Pts(1,2)-Pts(1,1));    % slope
for track
        bt = Pts(2,1) - mt*Pts(1,1);                    % y
intercept for track
        ms = (Pts(2,4)-Pts(2,3))/(Pts(1,4)-Pts(1,3));    % slope
for sealane
        bs = Pts(2,3) - ms*Pts(1,3);                    % y
intercept for sealane

        x_int = (bs-bt)/(mt-ms);                        % point
of cross over
        y_int = mt*x_int + bt;

        between1 = (Pts(1,1) <= x_int) ...
                & (x_int<= Pts(1,2)) ...
                & (Pts(2,1) <= y_int) ...
                & (y_int <= Pts(2,2));

        between2 = (Pts(1,1) >= x_int) ...

```

```

        & (x_int >= Pts(1,2)) ...
        & (Pts(2,1) >= y_int) ...
        & (y_int >= Pts(2,2));

    between3 = (Pts(1,1) >= x_int) ...
        & (x_int >= Pts(1,2)) ...
        & (Pts(2,1) <= y_int) ...
        & (y_int <= Pts(2,2));

    between4 = (Pts(1,1) <= x_int) ...
        & (x_int <= Pts(1,2)) ...
        & (Pts(2,1) >= y_int) ...
        & (y_int >= Pts(2,2));
    between = between1 | between2 | between3 | between4;
else
    between = false;
end

if between
    % If true, then we need to use this (x_int, y_int)
point in the area
    % calculations since the track and sealane cross each
other, i.e., this
    % is a real crossover point.

    % Need to calculate area with points Pts(:,1), Pts(:,3)
(counter track
    % points) and (x_int, y_int)
    CrossOverPt = [x_int; y_int];
    crossOver = true;
    thirdPt=CrossOverPt
else
    crossOver = false;
    % If false, then the track and sealane do not cross
each other
    thirdPt=0;
end

end

```

```

function [finalthirdPt, trackCntr, sealaneCntr, done] =
chooseThirdPtFromTandS(meanstart, maybes, track, sealane,
trackCntr, sealaneCntr, done)
% This functions determines whether the third point for
finding the area of
% triangle has to be selected from ships track, Sea lane,
or
% ships tracks and sea lane data points
% Function input: Likely next candidates for third data
points, Mean of first data point of Ship's track,
% Ship's track matrix, Sea lane matrix, and Sea lane, Sea
lane data point
% counter, Ship's track data point counter, flag to
indicate if all data
% points have finished.
% Fucntion output: Third point from either Sea lane or Ship
track, updated
% ships' track counter, updated sea lane counter, flag to
indicate if all
% data points have finished
% Created by: Tauseef Ashraf, LT CDR PN, September 2018
[~, trackCntrMax] = size(track);
    [~, sealaneCntrMax] = size(sealane);
    trackCand = cell2mat(maybes(1,2));
        sealaneCand = cell2mat(maybes(2,2));

    if trackCntrMax < trackCand
        dt = Inf;
    else
        dt = euDist(meanstart, track(:, trackCand));
    end

    if sealaneCntrMax < sealaneCand
        ds = Inf;
    else
        ds = euDist(meanstart, sealane(:, sealaneCand));
    end

    if ~and(dt == Inf, ds == Inf)
        if dt < ds
            finalthirdPt = {'track', trackCand};
            trackCntr = trackCntr+1;
        else
            finalthirdPt = {'sealane', sealaneCand};
            sealaneCntr = sealaneCntr+1;
        end
    end

```

```
        end
    else
        done= true;
        finalthirdPt = [0 0]';
    end
end
```

```

function [finalthirdPt, trackCntr, sealaneCntr, crossOver,
done] = determineThirdPt(twoPts, maybes, track, sealane,
meanstart, trackCntr, sealaneCntr, crossOver, thirdPt,
done);
% This functions determines the third point for calculating
triangular area enclosed by three points
% Function input: Two data points already determined for
new triangular
% area, likely next candidates for third data points,
Ship's track
% matrix, Sea lane matrix, Mean of first data point of
Ship's track and Sea
% lane, Sea lane data point counter, Ship's track data
point counter,
% Crossover flag, likely third point, flag to check if data
points of sea lane or ship's track
% have finished
% Fucntion output: Finalized third point of triangle,
Ship's track counter,
% Sea lane counter, Crossover flag, Crossover if there is a
cross over, flag
% to check if data points of sea lane or ship's track have
finished
% Created by: Tauseef Ashraf, LT CDR PN, September 2018
    if crossOver == true;
        finalthirdPt=thirdPt;
        crossOver=false;    %reset crossOver to false;
        checklastcrossover=1;
    else
        finalthirdPt=0;
        crossOver=false;
        checklastcrossover=0;
    end
    if ~checklastcrossover
        [thirdPt,crossOver] = doTheyCross(twoPts, maybes,
track, sealane, meanstart, trackCntr, sealaneCntr);

        if crossOver == true
            finalthirdPt=thirdPt;
        else
            [finalthirdPt, trackCntr, sealaneCntr, done] =
chooseThirdPtFromTandS(meanstart, maybes, track, sealane,
trackCntr, sealaneCntr, done);
        end
    end
end
end

```

```

% This program defines two sea lanes from ten simulated
ship's tracks
% using modified k-means algorithm discussed in Thesis. We
first generated
% ten ships' tracks, found area metric between all of them,
and based on
% modified k-means algorithm, clustered them in optimally
into two sea
% lanes. The sea lane is then defined by averaging all the
tracks belonging
% to each cluster(sea lane).
% Created by: Tauseef Ashraf, LT CDR PN, September 2018
clear all
close all
clc
SIZEOFDATA=20;           % size of total data points of Sea
lanes
TRACKS_PER_SEALANE = 5; % Number of Ships' tracks in each
Sea lane
NUM_SEALANES = 2;       % Number of total Sea lanes

AreaMatrix=zeros(TRACKS_PER_SEALANE*NUM_SEALANES,
NUM_SEALANES+1);
NUM_AREAS=((2*TRACKS_PER_SEALANE)^2-
(2*TRACKS_PER_SEALANE))/2;
    % Number of total distinct areas between ships tracks
sealane_fixes=zeros(2,SIZEOFDATA,NUM_SEALANES);
    % Matrix for storing Sea lane data points
track=zeros(2,SIZEOFDATA,NUM_SEALANES);
    % Matrix for ships' tracks data points
for area_index=1:TRACKS_PER_SEALANE;

sealane_fixes(:,:,1)=[linspace(0,10,SIZEOFDATA);linspace(0,
10,SIZEOFDATA)];
sealane_fixes(:,:,2)=[linspace(0,10,SIZEOFDATA);linspace(10
,0,SIZEOFDATA)];
% standard deviation for each ship's track
stan_dev = [0.05*ones(TRACKS_PER_SEALANE-3,1);
0.2*ones(2,1); 0.3];
% x and y-axis matrix for ships' tracks
[ tkcomset1_x, tkcomset1_y ] = makeTracks( stan_dev,
sealane_fixes(:,:,1),...
        SIZEOFDATA );
[ tkcomset2_x, tkcomset2_y ] = makeTracks( stan_dev,
sealane_fixes(:,:,2),...
        SIZEOFDATA );

```



```

tkcomset_x= [tkcomset1_x;tkcomset2_x];
tkcomset_y= [tkcomset1_y;tkcomset2_y];

for m=1:2*TRACKS_PER_SEALANE
    % points for first triangle
    twoPts = {'track', 1;'sealane', 1};
    trackCntr = 1;
    sealaneCntr = 1;
        for n=1:2*TRACKS_PER_SEALANE
            sealane=[tkcomset_x(m,:);tkcomset_y(m,:)];
            track=[tkcomset_x(n,:);tkcomset_y(n,:)];
            trackCntr = 1; % Ship's track data point
counter
            sealaneCntr = 1;% Sealane data point
counter
            % Average of first data points of Sea lane
and Ship's track
            meanstart = 0.5*(track(:,1) +
sealane(:,1));
            crossOver=false;finalthirdPt=0;
            done = false; % flag to check end of data
points
            totalArea = 0; %Total area between Ship's
track
                while ~done
                    [~,Ns,~]=size(sealane_fixes);
                    [~,Nt]=size(tkcomset_y(n,:));
                    twoPts = {'track', trackCntr;
% points for next triangle
                    'sealane', sealaneCntr};

                    candidate3rdPts = {'track',
trackCntr+1; ...
                    'sealane', sealaneCntr+1; ...
                    'crossover', 0};
                    [finalthirdPt, trackCntr,
sealaneCntr, crossOver, done] = determineThirdPt(twoPts,
candidate3rdPts, track, sealane, meanstart, trackCntr,
sealaneCntr, crossOver, finalthirdPt, done);

                    if done
                        break;
                    end
                    triangleArea = triA(twoPts,
finalthirdPt, track , sealane);

```

```

totalArea = totalArea +
abs(triangleArea);

                                %test to check if third point
was a cross over
                                if crossOver == true;
                                trackCntr= trackCntr+1;
%in case of cross over we select next two points
                                sealaneCntr= sealaneCntr+1;
                                else
                                trackCntr= trackCntr;
                                sealaneCntr= sealaneCntr;
                                end
                                end %end of while loop
finalarea(m,n)= totalArea;%final areas between all
tracks
                                end
                                end
                                end
% plotting all Ships' tracks
for x=1:2*TRACKS_PER_SEALANE
    plot(tkcomset_x(x,:),tkcomset_y(x,:),'.-');

                                hold on
end

%% Defining Sea lane from Area metric
M=0;

v=1:1:2*TRACKS_PER_SEALANE;
% Combination for selecting sea lanes from Ships tracks
C = nchoosek(v,2);

    for jj=1:length(C);
        M=0;
        for ii=1:2*TRACKS_PER_SEALANE;
            if
finalarea(ii,C(jj,1))<finalarea(ii,C(jj,2))
                result(ii,jj)=1;
                M=M+finalarea(ii,C(jj,1));
            else
                result(ii,jj)=2;
                M=M+finalarea(ii,C(jj,2));;
            end
        end
    end

```

```

                end
            end
            M_areas(jj,1)=M;
        end

[smallest_M,M_index]=mink(M_areas,1);
cluster=result(:,M_index)% Making Clusters of Ship's tracks

% sea-lanes found from modified k-means algorithm
sealaneA=[sum(tkcomset_x(1:TRACKS_PER_SEALANE,:))/TRACKS_PER_SEALANE;sum(tkcomset_y(1:TRACKS_PER_SEALANE,:))/TRACKS_PER_SEALANE];
sealaneB=[sum(tkcomset_x(TRACKS_PER_SEALANE+1:2*TRACKS_PER_SEALANE,:))/TRACKS_PER_SEALANE;sum(tkcomset_y(TRACKS_PER_SEALANE+1:2*TRACKS_PER_SEALANE,:))/TRACKS_PER_SEALANE];
hold on
plot(sealaneA(1,:),sealaneA(2,:),':','LineWidth',2)
plot(sealaneB(1,:),sealaneB(2,:),':','LineWidth',2)
xlim([-2,15]);ylim([-2,15]);
legend('Ship Track 1','Ship Track 2','Ship Track 3',...
        'Ship Track 4','Ship Track 5','Ship Track 6',...
        'Ship Track 7','Ship Track 8','Ship Track 9','Ship
Track 10',...
        'Sea lane A','Sea lane B','location','best');
xlabel('Distance (\textit{Nm})', ...
        'FontSize',12,'Color','k', 'Interpreter', 'LaTeX');
ylabel('Distance (\textit{Nm})', ...
        'FontSize',12,'Color','k', 'Interpreter', 'LaTeX');
title('Ship tracks in sea lane \textit{A} to \textit{B}',
...
        'FontSize',12,'Color','k', 'Interpreter', 'LaTeX')

```

```

% This program classifies ships tracks into Sea lanes using
two methods:
%     1. User defined threshold
%     2. Modified k-means algorithm
% As we already have informtion about the sea lanes from
previous code, we
% use the sea lane information to generate ten ships'
tracks, found area
% metric of all ships' tracks with respect to both the sea
lanes and
% classify these ships' tracks into either of sea lanes
using two different
% methods
% Created by: Tauseef Ashraf, LT CDR PN, September 2018

clear all
close all
clc
SIZEOFDATA=20;           % size of total data points of Sea
lanes
TRACKS_PER_SEALANE = 5; % Number of Ships' tracks in each
Sea lane
NUM_SEALANES = 2;       % Number of total Sea lanes

AreaMatrix=zeros(TRACKS_PER_SEALANE*NUM_SEALANES,
NUM_SEALANES+1);
NUM_AREAS=((2*TRACKS_PER_SEALANE)^2-
(2*TRACKS_PER_SEALANE))/2;
    % Number of total distinct areas between ships tracks
sealane_fixes=zeros(2,SIZEOFDATA,NUM_SEALANES);
    % Matrix for storing Sea lane data points
track=zeros(2,SIZEOFDATA,NUM_SEALANES);
    % Matrix for ships' tracks data points
for area_index=1:TRACKS_PER_SEALANE;
% Sea lanes 1 and 2
sealane_fixes(:, :, 1)=[linspace(0,10,SIZEOFDATA);linspace(0,
10,SIZEOFDATA)];
sealane_fixes(:, :, 2)=[linspace(0,10,SIZEOFDATA);linspace(10
,0,SIZEOFDATA)];
% standard deviation for each ship's track
stan_dev = [0.05*ones(TRACKS_PER_SEALANE-3,1);
0.2*ones(2,1); 0.3];
% x and y-axis matrix for ships' tracks
[ tkcomset1_x, tkcomset1_y ] = makeTracks( stan_dev,
sealane_fixes(:, :, 1), ...
        SIZEOFDATA );

```

```

[ tkcomset2_x, tkcomset2_y ] = makeTracks( stan_dev,
sealane_fixes(:, :, 2), ...
        SIZEOFDATA );
tkcomset_x= [tkcomset1_x;tkcomset2_x];
tkcomset_y= [tkcomset1_y;tkcomset2_y];

    for m=1:NUM_SEALANES;
        % points for first triangle
        twoPts = {'track', 1;'sealane', 1};
        trackCntr = 1;
        sealaneCntr = 1;
            for n=1:2*TRACKS_PER_SEALANE
                sealane=sealane_fixes(:, :, m);
                track=[tkcomset_x(n, :);tkcomset_y(n, :)];
                trackCntr = 1; % Ship's track data point
counter
                sealaneCntr = 1;% Sealane data point
counter
                % Average of first data points of Sea lane
and Ship's track
                meanstart = 0.5*(track(:,1) +
sealane(:,1));
                crossOver=false;finalthirdPt=0;
                done = false; % flag to check end of data
points
                totalArea = 0; %Total area between Ship's
track
                    while ~done
                        [~,Ns,~]=size(sealane_fixes);
                        [~,Nt]=size(tkcomset_y(n, :));
                        twoPts = {'track', trackCntr;
% points for next triangle
                        'sealane', sealaneCntr};

                        candidate3rdPts = {'track',
trackCntr+1; ...
                        'sealane', sealaneCntr+1; ...
                        'crossover', 0};
                        [finalthirdPt, trackCntr,
sealaneCntr, crossOver, done] = determineThirdPt(twoPts,
candidate3rdPts, track, sealane, meanstart, trackCntr,
sealaneCntr, crossOver, finalthirdPt, done);

                            if done
                                break;

```

```

                                end
                                triangleArea = triA(twoPts,
finalthirdPt, track , sealane);
                                totalArea = totalArea +
abs(triangleArea);

                                %test to check if third point
was a cross over
                                if crossOver == true;
                                trackCntr= trackCntr+1;
%in case of cross over we select next two points
                                sealaneCntr= sealaneCntr+1;
                                else
                                trackCntr= trackCntr;
                                sealaneCntr= sealaneCntr;
                                end
                                end %end of while loop
                                finalarea(n,m)= totalArea;%final areas between all
tracks

                                end
                                end
                                end
% plotting all Ships' tracks and Sea lanes

for y=1:NUM_SEALANES;

plot(sealane_fixes(1,:,y),sealane_fixes(2,:,y),':', 'LineWidth
th',3)
    hold on
end

for x=1:2*TRACKS_PER_SEALANE
    plot(tkcomset_x(x,:),tkcomset_y(x,),'.-');
end
xlim([-2,15]);ylim([-2,15]);
legend('Sea lane A', 'Sea lane B', 'Ship Track 1','Ship
Track 2','Ship Track 3',...
'Ship Track 4','Ship Track 5','Ship Track 6',...
'Ship Track 7','Ship Track 8','Ship Track 9','Ship
Track 10',...
'location','best');
xlabel('Distance (\textit{Nm})', ...
'FontSize',12,'Color','k', 'Interpreter', 'LaTeX');

```

```

ylabel('\textit{Distance (\textit{Nm})}', ...
       'FontSize',12,'Color','k', 'Interpreter', 'LaTeX');
title('Ship tracks in sea lane \textit{A} to \textit{B}',
      ...
      'FontSize',12,'Color','k', 'Interpreter', 'LaTeX')
%% Classifying ships' tracks into Sea lanes using User
defined threold
Thresh_SLA=5;
Thresh_SLB=8;
    for j=1:2*TRACKS_PER_SEALANE;
        if finalarea(j,1)<=Thresh_SLA
            Sealane_cluster(j,1)=1;
        else
            Sealane_cluster(j,1)=0;
        end

        if finalarea(j,2)<=Thresh_SLB
            Sealane_cluster(j,2)=2;
        else
            Sealane_cluster(j,2)=0;
        end
    end
final_sealane_cluster=Sealane_cluster(:,1)+Sealane_cluster(
:,2);
    % Notice there may be instances when a particular ships
tracks has not
    % been classified into any of the sea lanes

%% Classifying ships' tracks into Sea lanes using modified
k-means algorithm
% We call it modified k-means because this takes into
account the area
% metric with respect to both sea lanes and classify ships'
tracks into Sea
% lanes based on minimum area metric with respect to each
Sea lane. Hence,
% it will forcefully classify all ships' tracks into either
of the two Sea
% lanes.
for j=1:2*TRACKS_PER_SEALANE;
    if finalarea(j,1)<finalarea(j,2)
        Sealane_cluster_kmeans(j,1)=1;
    else
        Sealane_cluster_kmeans(j,1)=2;
    end
end
end

```

```

% This program calculates the probability of Ships' tracks
being in two sea
% lanes. We assume that we already have information about
the sea lane using
% the previous code. We generated sixteen ship tracks on
these sea lanes
% and found area metric of these ships tracks with respect
to each sea lane. We calculate the parameters of ships
tracks area metric PDFs( Means & Variances) in both sea
lanes.
% We then define nine ship tracks as test set and find the
probability of
% each of these ships tracks being in either of the sea
lanes defined earlier.
% Created by: Tauseef Ashraf, LT CDR PN, September 2018
clear all
close all
clc
SIZEOFDATA=20;
TRACKS_PER_SEALANE = 8;
NUM_SEALANES = 2;
AreaMatrix=zeros(TRACKS_PER_SEALANE,2*NUM_SEALANES);
AreaMatrix=zeros(TRACKS_PER_SEALANE*NUM_SEALANES,
NUM_SEALANES+1);
    % use the last column to indicate which sealane the
track is from.
sealane_fixes=zeros(2,SIZEOFDATA,NUM_SEALANES);
    % each fix has X and Y. Each track has SIZEOFDATA
fixes.
track=zeros(2,SIZEOFDATA,NUM_SEALANES);

sealane_endpoints = [0 20; 20 0];
for sealane_index=1:NUM_SEALANES^2;

    % Area Matrix 1 - Sealane 1 - Tracks 1
    if sealane_index==1
        points =
linspace(sealane_endpoints(sealane_index,1), ...
        sealane_endpoints(sealane_index,2),
SIZEOFDATA);
        sealane_fixes(:, :, sealane_index)=[ points; points
];
        clear points;

        figure(1)

```



```

        plot(sealane_fixes(1,:,1),sealane_fixes(2,:,1),'--
*');
        hold on
        legend('Sea Lane A','location','best');
        title('Ships Tracks in Sea lanes');

        stan_dev = [0.1*ones(TRACKS_PER_SEALANE-4,1);
0.2*ones(2,1); 0.5*ones(2,1)];

        [ tkcomset1_x, tkcomset1_y ] = makeTracks(
stan_dev, sealane_fixes(:, :, 1), ...
        SIZEOFDATA );

        % [sizeoftrackset1,~]=size(tkcomset1_x);
        hold on
        for n=1:TRACKS_PER_SEALANE
            plot(tkcomset1_x(n,:),tkcomset1_y(n:),'.-
');
            legend('sealane','track number 1','track
number 2','track number 3',...
            'track number 4','track number 5',
            'track number 6',...
            'track number 7', 'track number
8','track number 9', 'track number 10',...
            'track number 11','track number 12',
            'track number 13');
            hold on
            twoPts = {'track', 1;          % points for
first triangle
            'sealane', 1};

            trackCntr = 1;
            sealaneCntr = 1;

            track=[tkcomset1_x(n,:);tkcomset1_y(n,:)];
            sealane=sealane_fixes(:, :, 1);
            meanstart = 0.5*(track(:,1) +
sealane(:,1));
            % meanstart = 0.5*(tkcomset1_x(n,:) +
sealane_fixes(1,:,1));
            crossOver=false;finalthirdPt=0;
            % three points make a triangle. First
candidate is always...
            ...from track and second is always from
sealane.

            done = false;

```

```

totalArea = 0;
    while ~done
        [~,Ns,~]=size(sealane_fixes);
        [~,Nt]=size(tkcomset1_y(n,:));
        twoPts = {'track', trackCntr;
% points for next triangle
                'sealane', sealaneCntr};

        candidate3rdPts = {'track',
trackCntr+1;    ...
                        'sealane', sealaneCntr+1; ...
                        'crossover', 0};
        [finalthirdPt, trackCntr,
sealaneCntr, crossOver, done] = determineThirdPt(twoPts,
candidate3rdPts, track, sealane, meanstart, trackCntr,
sealaneCntr, crossOver, finalthirdPt, done);

                if done
                    break;
                end
                triangleArea = triA(twoPts,
finalthirdPt, track , sealane);
                totalArea = totalArea +
abs(triangleArea);

                %test to check if third point was a cross over
                    if crossOver == true;
                        trackCntr= trackCntr+1;    %in
case of cross over we select next two points
                        sealaneCntr= sealaneCntr+1;
                    else
                        trackCntr= trackCntr;
                        sealaneCntr= sealaneCntr;
                    end

                end    %end of while loop
        finalarea(n,1)= totalArea;
    end

AreaMatrix(1:TRACKS_PER_SEALANE,sealane_index)=finalarea;

    AreaMatrix(1:TRACKS_PER_SEALANE,3)=1;

% Area Matrix 2 - Selane 2 - Tracks 2

```

```

elseif sealane_index==2

    points =
linspace(sealane_endpoints(sealane_index,1), ...
        sealane_endpoints(sealane_index,2),
SIZEOFDATA);
    sealane_fixes(:, :, sealane_index)=[ points;
fliplr(points) ];
    clear points;

plot(sealane_fixes(1, :, sealane_index), sealane_fixes(2, :, sea
lane_index), '--*');
    legend('Sea Lane B', 'location', 'best');
    title('Training data set of Ships Tracks in Sea
lanes');

    [ tkcomset2_x, tkcomset2_y ] = makeTracks (
stan_dev, sealane_fixes(:, :, 2), ...
        SIZEOFDATA );

    hold on
        for n=1:TRACKS_PER_SEALANE

plot(tkcomset2_x(n, :), tkcomset2_y(n, :), '-. ');
            hold on
            twoPts = {'track', 1;           %
points for first triangle
                'sealane', 1};

            trackCntr = 1;
            sealaneCntr = 1;

track=[tkcomset2_x(n, :); tkcomset2_y(n, :)];
            sealane=sealane_fixes(:, :, 2);

            meanstart = 0.5*(track(:, 1) +
sealane(:, 1));

            crossOver=false; finalthirdPt=0;

```

```

                                % three points make a triangle.
First candidate is always from track and second is always
from sealane.
                                done = false;
                                totalArea = 0;
                                while ~done

[~,Ns,~]=size(sealane_fixes);

[~,Nt]=size(tkcomset2_y(n,:));
                                twoPts = {'track',
trackCntr;          % points for next triangle
                                'sealane', sealaneCntr};

                                candidate3rdPts = {'track',
trackCntr+1;      ...
                                'sealane',
sealaneCntr+1;  ...
                                'crossover', 0};
                                [finalthirdPt, trackCntr,
sealaneCntr, crossOver, done] = determineThirdPt(twoPts,
candidate3rdPts, track, sealane, meanstart, trackCntr,
sealaneCntr, crossOver, finalthirdPt, done);

                                if done
                                    break;
                                end
                                triangleArea =
triA(twoPts, finalthirdPt, track , sealane);
                                totalArea = totalArea +
abs(triangleArea);

                                if crossOver == true;
                                    trackCntr=
trackCntr+1;          %in case of cross over we select next two
points
                                    sealaneCntr=
sealaneCntr+1;

                                else
                                    trackCntr=
trackCntr;
                                    sealaneCntr=
sealaneCntr;
                                end

```



```

        % three points make a triangle. First candidate
is always from track and second is always from sealane.
        done = false;
        totalArea = 0;
        while ~done
            [~,Ns,~]=size(sealane_fixes);
            [~,Nt]=size(tkcomset1_y(n,:));
            twoPts = {'track', trackCntr;           %
points for next triangle
                    'sealane', sealaneCntr};

            candidate3rdPts = {'track',
trackCntr+1;    ...
                            'sealane', sealaneCntr+1; ...
                            'crossover', 0};
            [finalthirdPt, trackCntr, sealaneCntr,
crossOver, done] = determineThirdPt(twoPts,
candidate3rdPts, track, sealane, meanstart, trackCntr,
sealaneCntr, crossOver, finalthirdPt, done);
            if done
                break;
            end
            triangleArea = triA(twoPts,
finalthirdPt, track , sealane);
            totalArea = totalArea +
abs(triangleArea);

            %test to check if third point was a cross over
            if crossOver == true;
                trackCntr= trackCntr+1;           %in
case of cross over we select next two points
                sealaneCntr= sealaneCntr+1;
            else
                trackCntr= trackCntr;
                sealaneCntr= sealaneCntr;
            end

        end           %end of while loop
        finalarea(n,1)= totalArea;
    end

    AreaMatrix(1:TRACKS_PER_SEALANE,2)=finalarea;
end

```

```

end

legend('Sea lane A','Ship track 1','Ship track 2',...
'Ship track 3', 'Ship track 4', 'Ship track 5', 'Ship track
6','Ship track 7','Ship track 8',...
'Sea lane B','Ship track 9', 'Ship track 10', 'Ship track
11','Ship track 12','Ship track 13','Ship track 14','Ship
track 15',...
'Ship track 16','location','northwest');
xlabel('Distance (\textit{Nm})', ...
'FontSize',12,'Color','k', 'Interpreter', 'LaTeX');
ylabel('Distance (\textit{Nm})', ...
'FontSize',12,'Color','k', 'Interpreter', 'LaTeX');
title('Training data of Ship tracks in Sea lane \textit{A}
and \textit{B}', ...
'FontSize',12,'Color','k', 'Interpreter', 'LaTeX')

%% Finding the PDF parameters (Means and Standard
Deviation) from Training data
figure (2)
SLAtracks=AreaMatrix(:,1);
SLBtracks=AreaMatrix(:,2);
meanTKBSLA=mean(SLAtracks(TRACKS_PER_SEALANE+1:2*TRACKS_PER
_SEALANE,1));
meanTKASLB=mean(SLBtracks(1:TRACKS_PER_SEALANE,1));
meanTKASLA=0;
meanTKBSLB=0;
% Standard Deviation
SD_tracks_TKA_SLA =
standard_deviation(0,SLAtracks(1:TRACKS_PER_SEALANE,1));
SD_tracks_TKB_SLA = standard_deviation(meanTKBSLA,
SLAtracks(TRACKS_PER_SEALANE+1:2*TRACKS_PER_SEALANE,1));
SD_tracks_TKB_SLB = standard_deviation(0,
SLBtracks(TRACKS_PER_SEALANE+1:2*TRACKS_PER_SEALANE,1));
SD_tracks_TKA_SLB= standard_deviation(meanTKASLB,
SLBtracks(1:TRACKS_PER_SEALANE,1));

%% Generating Test Set

plot(sealane_fixes(1,:,1),sealane_fixes(2,:,1),'-
.', 'LineWidth',2);
hold on

```



```

plot(sealane_fixes(1,:,2),sealane_fixes(2,:,2),'-
.', 'LineWidth',2);
%           title('Testing data set of Ships Tracks in
Sea lanes');
%           legend('Sea Lane A','Sea Lane B','Ship
1','Ship 2','Ship 3','Ship 4',...
%           'Ship 5','Ship 6','Ship 7','Ship 8','Ship
9''location','best')
           testsetx=[linspace(2, 18, 20);linspace(4, 16,
20);linspace(6, 14, 20);linspace(8, 12, 20);linspace(10,
10, 20);...
           linspace(12, 8, 20);linspace(14, 6,
20);linspace(16, 4, 20);linspace(18,2, 20)];
           testsety=[linspace(0, 20, 20);linspace(0, 20,
20);linspace(0, 20, 20);linspace(0, 20, 20);linspace(0, 20,
20);...
           linspace(0, 20, 20);linspace(0, 20,
20);linspace(0, 20, 20);linspace(0, 20, 20)];
           hold on
           [num,~]=size(testsetx);
           for ii=1:num
           plot(testsetx(ii,:),testsety(ii,),'.-')
           end
           xlim([-2,25]);ylim([-2,25]);
legend('Sea lane A', 'Sea lane B', 'Ship Track 1','Ship
Track 2','Ship Track 3',...
'Ship Track 4','Ship Track 5','Ship Track 6',...
'Ship Track 7','Ship Track 8','Ship Track 9',...
'location','best');
xlabel('Distance (\textit{Nm})', ...
'FontSize',12,'Color','k', 'Interpreter', 'LaTeX');
ylabel('Distance (\textit{Nm})', ...
'FontSize',12,'Color','k', 'Interpreter', 'LaTeX');
title('Test Set of Ship tracks in sea lane \textit{A} to
\textit{B}', ...
'FontSize',12,'Color','k', 'Interpreter', 'LaTeX')

% Finding area metric between ships' tracks test set and
Sea lanes A and B

SIZEOFDATA=20;
TRACKS_IN_SEALANE= 9;
NUM_SEALANES = 2;

```

```

TestAreaMatrix=zeros(TRACKS_IN_SEALANE, NUM_SEALANES);

%-----
-%
    for n=1:TRACKS_IN_SEALANE
        twoPts = {'track', 1;           % points for
first triangle  'sealane', 1};

        trackCntr = 1;
        sealaneCntr = 1;

        track=[testsetx(n,:);testsety(n,:)];
        sealane=sealane_fixes(:, :,1);
        meanstart = 0.5*(track(:,1) +
sealane(:,1));

        crossOver=false;finalthirdPt=0;
        % three points make a triangle. First
candidate is always...
        ...from track and second is always from
sealane.

        done = false;
        totalArea = 0;
        while ~done
            [~,Ns,~]=size(sealane_fixes);
            [~,Nt]=size(testsetx(n,:));
            twoPts = {'track', trackCntr;
% points for next triangle  'sealane', sealaneCntr};

            candidate3rdPts = {'track',
trackCntr+1;   ...
                            'sealane', sealaneCntr+1; ...
                            'crossover', 0};
            [finalthirdPt, trackCntr,
sealaneCntr, crossOver, done] = determineThirdPt(twoPts,
candidate3rdPts, track, sealane, meanstart, trackCntr,
sealaneCntr, crossOver, finalthirdPt, done);

            if done
                break;
            end

            triangleArea = triA(twoPts,
finalthirdPt, track , sealane);
            totalArea = totalArea +
abs(triangleArea);

```

```

                                %test to check if third point was a
cross over
                                if crossOver == true;
                                trackCntr= trackCntr+1;          %in
case of cross over we select next two points
                                sealaneCntr= sealaneCntr+1;
                                else
                                trackCntr= trackCntr;
                                sealaneCntr= sealaneCntr;
                                end
                                end          %end of while loop
                                finalarea(n,1)= totalArea;
                                end
                                TestAreaMatrix(1:TRACKS_IN_SEALANE,1)=finalarea;

%-----%
% Area Matrix 2 - Selane 2 - Tracks 2

                                % Area Matrix 1 - Selane 1 -
Tracks 1
                                for n=1:TRACKS_IN_SEALANE
                                twoPts = {'track', 1;          % points for
first triangle
                                'sealane', 1};

                                trackCntr = 1;
                                sealaneCntr = 1;

                                track=[testsetx(n,:);testsety(n,:)];
                                sealane=sealane_fixes(:, :, 2);
                                meanstart = 0.5*(track(:,1) +
sealane(:,1));
                                crossOver=false;finalthirdPt=0;
                                % three points make a triangle. First
candidate is always...
                                ...from track and second is always from
sealane.
                                done = false;
                                totalArea = 0;
                                while ~done
                                [~,Ns,~]=size(sealane_fixes);
                                [~,Nt]=size(testsetx(n,:));

```

```

                                twoPts = {'track', trackCntr;
% points for next triangle      'sealane', sealaneCntr};

                                candidate3rdPts = {'track',
trackCntr+1;    ...
                                'sealane', sealaneCntr+1; ...
                                'crossover', 0};
                                [finalthirdPt, trackCntr,
sealaneCntr, crossOver, done] = determineThirdPt(twoPts,
candidate3rdPts, track, sealane, meanstart, trackCntr,
sealaneCntr, crossOver, finalthirdPt, done);

                                if done
                                    break;
                                end
                                triangleArea = triA(twoPts,
finalthirdPt, track , sealane);
                                totalArea = totalArea +
abs(triangleArea);

                                %test to check if third point was a cross over
                                if crossOver == true;
case of cross over we select next two points
                                trackCntr= trackCntr+1;    %in
                                sealaneCntr= sealaneCntr+1;
                                else
                                trackCntr= trackCntr;
                                sealaneCntr= sealaneCntr;
                                end

                                end    %end of while loop
                                finalarea(n,1)= totalArea;
                                end
                                TestAreaMatrix(1:TRACKS_IN_SEALANE,2)=finalarea;

%% For finding Probability of Test set
% Since some of the numbers are very tiny to be processed
by computer, we
% used the logarithmic form of PDF function.
SLAtracks=TestAreaMatrix(:,1); % Area Metric of Ships'
tracks belonging to
% to Sea lane A
SLBtracks=TestAreaMatrix(:,2);% Area Metric of Ships'
tracks belonging to

```

```

% to Sea lane B

log_PDF_SLA_TKA=log(2/sqrt(2*pi*SD_tracks_TKA_SLA^2))-
(((SLAtracks-0).^2)/(2*SD_tracks_TKA_SLA^2));
log_PDF_SLA_TKB=log(1/sqrt(2*pi*SD_tracks_TKB_SLA^2))-
((SLBtracks-meanTKBSLA).^2/(2*SD_tracks_TKB_SLA^2));
log_PDF_SLB_TKB=log(2/sqrt(2*pi*SD_tracks_TKB_SLB^2))-
((SLBtracks-0).^2/(2*SD_tracks_TKB_SLB^2));
log_PDF_SLB_TKA=log(1/sqrt(2*pi*SD_tracks_TKA_SLB^2))-
((SLAtracks-meanTKASLB).^2/(2*SD_tracks_TKA_SLB^2));
% PDF for Area metric of ships tracks belonging to Sea lane
A
log_PDF_Tracks_SLA=log_PDF_SLA_TKA+log_PDF_SLA_TKB;
% PDF for Area metric of ships tracks belonging to Sea lane
B
log_PDF_Tracks_SLB=log_PDF_SLB_TKB+log_PDF_SLB_TKA;
% Probability of Ship track being in Sea lane A
Pr_SLA_given_area=1./(1+exp(log_PDF_Tracks_SLB-
log_PDF_Tracks_SLA));
% Probability of Ship track being in Sea lane B
Pr_SLB_given_area=1./(1+exp(log_PDF_Tracks_SLA-
log_PDF_Tracks_SLB));

Prob_Testset=[Pr_SLA_given_area Pr_SLB_given_area];

```

```

% This program detects the group of ships following or
paralleling while underway
% using modified k-means algorithm discussed in Thesis. We
first generated
% six ships' tracks, found area metric between all of them,
and based on
% user defined threshold for area metric for maximum
allowable distance between ships
% tracks, we detect ships following one another in a Sea
lane. This program
% takes into account the temporal locality in area metric
by introducing
% third dimension in ships tracks information which
represents time
% information.
% Created by: Tauseef Ashraf, LT CDR PN, September 2018
clear all
close all
clc

sl=[linspace(0,10,11);linspace(0,10,11);linspace(10,20,11)]
;
% figure(1)
% plot3(sl(1,:),sl(2,:),sl(3,:), '--*');
% legend('Sea Lane','location','best');
%title('Sea lane and Tracks b/w Port A and Port B');

noise1=(randn(2,11)); %dummy vector for noise x-
axis of which decides how many tracks we will have
[ sizeofnoise1,~]=size(noise1);

for n=1:sizeofnoise1
noisefl1_x(n,:)=conv((randn(1,11)/50),ones(1,200));
%gaussian noise filtered for adding to x-axis of tracks
tkset1_x(n,:)=sl(1,:)+(noisefl1_x(n,1:11));% x-axis of set
of 5 tracks - trackset-1 with more filtered noise
noisefl1_y(n,:)=conv((randn(1,11)/50),ones(1,200));
%%gaussian noise filtered for adding to y-axis of tracks
tkset1_y(n,:)=sl(2,:)+(noisefl1_y(n,1:11));% y-axis of set
of 5 tracks - trackset-1 with more filtered noise
noisefl1_z(n,:)=conv((randn(1,11)/50),ones(1,200));
%%gaussian noise filtered for adding to z-axis of tracks
tkset1_z(n,:)=sl(3,:)+(noisefl1_z(n,1:11));% z-axis of set
of 5 tracks - trackset-1 with more filtered noise
end

```

```

noise1=(randn(2,11)); %dummy vector for noise x-
axis of which decides how many tracks we will have
[ sizeofnoise1,~]=size(noise1);

for n=1:sizeofnoise1
noisefl2_x(n,:)=conv((randn(1,11)/5),ones(1,200));
%gaussian noise filtered for adding to x-axis of tracks
tkset2_x(n,:)=sl(1,:)+(noisefl2_x(n,1:11));% x-axis of set
of 5 tracks - trackset-1 with more filtered noise
noisefl2_y(n,:)=conv((randn(1,11)/5),ones(1,200));
%%gaussian noise filtered for adding to x-axis of tracks
tkset2_y(n,:)=sl(2,:)+(noisefl2_y(n,1:11));% y-axis of set
of 5 tracks - trackset-1 with more filtered noise
noisefl2_z(n,:)=conv((randn(1,11)/5),ones(1,200));
%%gaussian noise filtered for adding to z-axis of tracks
tkset2_z(n,:)=sl(3,:)+(noisefl2_z(n,1:11));% z-axis of set
of 5 tracks - trackset-1 with more filtered noise
end

noise1=(randn(2,11)); %dummy vector for noise x-
axis of which decides how many tracks we will have
[ sizeofnoise1,~]=size(noise1);

for n=1:sizeofnoise1
noisefl3_x(n,:)=conv((randn(1,11)/2),ones(1,200));
%gaussian noise filtered for adding to x-axis of tracks
tkset3_x(n,:)=sl(1,:)+(noisefl3_x(n,1:11));% x-axis of set
of 2 tracks - trackset-1 with more filtered noise
noisefl3_y(n,:)=conv((randn(1,11)/2),ones(1,200));
%%gaussian noise filtered for adding to y-axis of tracks
tkset3_y(n,:)=sl(2,:)+(noisefl3_y(n,1:11));% y-axis of set
of 2 tracks - trackset-1 with more filtered noise
noisefl3_z(n,:)=conv((randn(1,11)/2),ones(1,200));
%%gaussian noise filtered for adding to z-axis of tracks
tkset3_z(n,:)=sl(3,:)+(noisefl2_z(n,1:11));% z-axis of set
of 2 tracks - trackset-1 with more filtered noise
end
tkcomset_x=[tkset1_x;tkset2_x;tkset3_x]; %making a
single combined vector of x-axis of all tracks
tkcomset_y=[tkset1_y;tkset2_y;tkset3_y]; %making a
single combined vector of y-axis of all tracks
tkcomset_z=[tkset1_z;tkset2_z;tkset3_z]; %making a
single combined vector of z-axis of all tracks
[ sizeoftrackset1,~]=size(tkcomset_x);
hold on

```

```

for nn=1:sizeoftrackset1    %only for plotting all the
tracks
plot3(tkcomset_x(nn,:),tkcomset_y(nn,:),tkcomset_z(nn,:),'-
*');
    grid on
end
legend('Ship track 1','Ship track 2','Ship track 3','Ship
track 4',...
'Ship track 5','Ship track 6','location','best');
xlabel('\Distance (\textit{Nm})', ...
'FontSize',12,'Color','k', 'Interpreter', 'LaTeX');
ylabel('\Distance (\textit{Nm})', ...
'FontSize',12,'Color','k', 'Interpreter', 'LaTeX');
zlabel('\Arbitrary Time unit', 'FontSize',12,'Color','k',
'Interpreter', 'LaTeX')
title('Ship tracks in a sea lane',...
'FontSize',12,'Color','k', 'Interpreter', 'LaTeX')

for m=1:sizeoftrackset1

%sealane=[tkcomset_x(m,:);tkcomset_y(m,:);tkcomset_z(m,:)];
    for n=1:sizeoftrackset1

        trackCntr = 1;
        sealaneCntr = 1;

%plot3(tkcomset_x(n,:),tkcomset_y(n,:),tkcomset_z(n,:));
        %grid on

track=[tkcomset_x(n,:);tkcomset_y(n,:);tkcomset_z(n,:)];

sealane=[tkcomset_x(m,:);tkcomset_y(m,:);tkcomset_z(m,:)];
    meanstart_x = 0.5*(tkcomset_x(n,1) + sealane(1,1));
    meanstart_y = 0.5*(tkcomset_y(n,1) + sealane(2,1));
    meanstart_z = 0.5*(tkcomset_z(n,1) + sealane(3,1));
    meanstart=[meanstart_x;meanstart_y;meanstart_z];
    crossOver=false;finalthirdPt=0;
    done = false;
    totalArea = 0;
    while ~done
        twoPts = {'track', trackCntr;           % points for next
triangle
'sealane', sealaneCntr};

        candidate3rdPts = {'track', trackCntr+1;   ...
'sealane', sealaneCntr+1; ...

```



```

        'crossover', 0};
    [finalthirdPt, trackCntr, sealaneCntr, crossOver, done]
= determineThirdPt(twoPts, candidate3rdPts, track, sealane,
meanstart, trackCntr, sealaneCntr, crossOver, finalthirdPt,
done);

    if done

        break;
    end
    triangleArea = triA(twoPts, finalthirdPt, track ,
sealane)
    totalArea = totalArea + abs(triangleArea);
    %test to check if third point was a cross over
    if crossOver == true;
        trackCntr= trackCntr+1;      %in case of cross
over we select next two points
        sealaneCntr= sealaneCntr+1;
    else
        trackCntr= trackCntr;
        sealaneCntr= sealaneCntr;
    end

    end      %end of while loop
    finalarea(m,n)= totalArea;
end
end
function [finalthirdPt, trackCntr, sealaneCntr, crossOver,
done] = determineThirdPt(twoPts, maybes, track, sealane,
meanstart, trackCntr, sealaneCntr, crossOver, thirdPt,
done);
    [finalthirdPt, trackCntr, sealaneCntr, done] =
chooseThirdPtFromTandS(meanstart, maybes, track, sealane,
trackCntr, sealaneCntr, done);
end

```

LIST OF REFERENCES

- [1] International Maritime Organization. (n.d.). Introduction to IMO. [Online]. Available: <http://www.imo.org/en/About/Pages/Default.aspx>. Accessed May 18, 2018.
- [2] International Maritime Organization. (n.d.). Regulations for carriage of AIS. [Online]. Available: <http://www.imo.org/en/OurWork/safety/navigation/pages/ais.aspx>. Accessed Aug. 7, 2018.
- [3] Fentropy Marine Services. (n.d.). AIS –Equip your boat with AIS today. [Online]. Available: <http://www.fentropy.com/wp/navigation-solutions/ais/>. Accessed Aug. 6, 2018.
- [4] R. O. Lane, D. A. Nevell, S. D. Hayward, and T. W. Beaney, “Maritime anomaly detection and threat assessment,” in *Proc. of the Int. Conf. on Inf. Fus.*, Edinburgh, 2010. [Online]. doi: 10.1109/ICIF.2010.5711998
- [5] D. Garagic, B. J. Rhodes, N. A. Bomberger, and M. Zandipour, “Adaptive mixture-based neural network approach for higher-level fusion and automated behavior monitoring,” in *Proc. of Int. Conf. on Comm.*, Dresden, 2009. [Online]. doi: 10.1109/ICC.2009.5198703
- [6] R. Laxhammar, G. Falkman, and E. Sviestins, “Anomaly detection in sea traffic - A comparison of the Gaussian mixture model and the kernel density estimator,” in *Proc. of Int. Conf. on Inf. Fus.*, Seattle, 2009. [Online]. Available: <http://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=5203766&isnumber=5203583>
- [7] H. Shao, N. Japkowicz, R. Abielmona, and R. Falcon, “Vessel track correlation and association using fuzzy logic and echo state networks,” in *Proc. of IEEE Cong. on Evol. Comp.*, Beijing, 2014, [Online]. doi: 10.1109/CEC.2014.6900231
- [8] A. McAbee, “Traffic pattern detection using the Hough transformation for anomaly detection to improve maritime domain awareness,” M.S. thesis, Dept. of Elec. and Comp. Eng., NPS, Monterey, CA, USA, 2013. [Online]. Available: https://calhoun.nps.edu/bitstream/handle/10945/38977/13Dec_McAbee_Ashley.pdf?sequence=1&isAllowed=y
- [9] K. Tester, “A spatiotemporal clustering approach to maritime domain awareness,” M.S. thesis, Dept. of Elec. and Comp. Eng., NPS, Monterey, CA, USA, 2013. [Online]. Available: https://calhoun.nps.edu/bitstream/handle/10945/37731/13Sep_Tester_Kristofer.pdf?sequence=1&isAllowed=y

- [10] U.S. Dept. of Defense. (2007, Mar. 22). Department of Defense Dictionary of Military and Associated Terms. [Online]. Available: <http://www.dtic.mil/dtic/tr/fulltext/u2/a469271.pdf>
- [11] W. L. Sharp, *Command and Control for Joint Maritime Operations*, JP 3–32, U.S. Dept. of Defense, Washington, DC, 2006. [Online]. Available: https://www.bits.de/NRANEU/others/jp-doctrine/jp3_32%2806%29.pdf
- [12] W. B. Carpenter, “Maritime domain awareness,” tactical memorandum, U.S. Dept. of the Navy, Norfolk, VA, USA, 2010. [Online]. Available: https://community.apan.org/wg/cne-cna-c6f-mda/m/mda_documents_of_interest/97945
- [13] S. C. Boraz, “Maritime domain awareness: Myths and realities,” *Nav. War Coll. Rev.*, vol. 62, no. 3, August 2007. [Online]. Available: www.dtic.mil/get-tr-doc/pdf?AD=ADA519318
- [14] J. T. Conway, G. Roughead, and T. T. Allen, “A cooperative strategy for 21st century sea power,” U.S. Dept. of Defense, Washington, DC, 2007. [Online]. Available: <http://www.dtic.mil/dtic/tr/fulltext/u2/a473237.pdf>
- [15] M. G. Mullen, “Navy maritime domain awareness concept,” official memorandum, U.S. Dept. of the Navy, Washington, DC, USA, 2007. [Online]. Available: <https://www.hsdl.org/?view&did=719590>
- [16] E. Martineau and J. Roy, “Maritime anomaly detection: Domain introduction and review of selected literature,” technical memorandum, Defence R&D Canada, Valcartier, Quebec, Canada, 2011. [Online]. Available: 2
- [17] Math Works. (n.d.). Unsupervised learning. [Online]. Available: <https://www.mathworks.com/discovery/unsupervised-learning.html>. Accessed Aug. 6, 2018.
- [18] U.S. Coast Guard. (n.d.) AIS requirements. [Online]. Available: <https://www.navcen.uscg.gov/?pageName=AISRequirementsRev>. Accessed Aug. 6, 2018.
- [19] International Telecommunications Union. (n.d.) Technical characteristics for an automatic identification system using time division multiple access in the VHF maritime mobile band, Recommendation ITU-R M.1371-5. [Online]. Available: http://www.itu.int/dms_pubrec/itu-r/rec/m/R-REC-M.1371-5-201402-I!!PDF-E.pdf. Accessed Aug. 6, 2018.
- [20] U.S. Coast Guard. (n.d.). Class A AIS position report. [Online]. Available: <https://www.navcen.uscg.gov/?pageName=AIMessagesA>. Accessed Aug. 6, 2018.

- [21] Shai Shalev-Shwartz and Shai Ben-David, *Understanding Machine Learning: From Theory to Algorithm*. Cambridge, UK: Cambridge University Press, pp. 268–271, 2014.
- [22] Z. Fu, W. Hu and T. Tan, “Similarity based vehicle trajectory clustering and anomaly detection,” in *IEEE Int. Conf. on Image Proc.* 2005, [Online].doi: 10.1109/ICIP.2005.153012
- [23] H. Li, J. Liu, R. Liu, N. Xiong, K. Wu, and T. Kim, “A Dimensionality Reduction-Based Multi-Step Clustering Method for Robust Vessel Trajectory Analysis,” *Sensors*, vol. 17, no. 8, p. 1792, Aug. 2017. [Online]. doi:10.3390/s17081792
- [24] A. R. Cohen and P. M. B. Vitányi, “Normalized compression distance of multisets with applications,” *IEEE Trans. on Pattern Analysis and Mach. Intel.*, vol. 37, no. 8, pp. 1602–1614, Aug. 2015. [Online]. doi: 10.1109/TPAMI.2014.2375175
- [25] P. M. B. Vitanyi, “Information distance in multiples,” *IEEE Trans. on Inf. Th.*, vol. 57, no. 4, pp. 2451–2456, Apr. 2011. [Online]. doi:10.1109/TIT.2011.2110130
- [26] Math Warehouse. (n.d.). Heron’s formula. [Online]. Available: <http://www.mathwarehouse.com/geometry/triangles/area/herons-formula-triangle-area.php>. Accessed Aug. 3, 2018.
- [27] Math World. (n.d.). Distance formula. [Online]. Available: <http://mathworld.wolfram.com/Distance.html>. Accessed Aug. 3, 2018.
- [28] C. W. Therrien and M. Tummala, *Probability and Random Process for Electrical and Computer Engineering*, 2nd ed., Boca Raton, FL, USA: CRC Press, 2012.

THIS PAGE INTENTIONALLY LEFT BLANK

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center
Ft. Belvoir, Virginia
2. Dudley Knox Library
Naval Postgraduate School
Monterey, California