



# Uncovering Hidden Concepts from AIS Data: A Network Abstraction of Maritime Traffic for Anomaly Detection

Ioannis Kontopoulos<sup>(✉)</sup>, Iraklis Varlamis<sup>(✉)</sup>, and Konstantinos Tserpes<sup>(✉)</sup>

Department of Informatics and Telematics, Harokopio University of Athens,  
Athens, Greece

{kontopoulos,varlamis,tserpes}@hua.gr  
<http://www.dit.hua.gr/>

**Abstract.** The compulsory use of Automatic Identification System (AIS) for many vessel types, which has been enforced by naval regulations, has opened new opportunities for maritime surveillance. AIS transponders are rich sources of information that everyone can collect using an RF receiver and provide real-time information about vessels' position. Properly taking advantage of AIS data, can uncover potential illegal behavior, offer real-time alerts and notify the authorities for any kind of anomalous vessel behavior. In this article, we extend an existing network abstraction of maritime traffic, that is based on nodes (called way-points) that correspond to naval areas of long stays or major turns for vessels (e.g. ports, capes, offshore platforms etc.) and edges (called traversals) that correspond to the routes followed by vessels between two consecutive way-points. The current work, focuses on the connections of this network abstraction and enriches them with semantic information about the different ways that vessels employ when traversing an edge. For achieving this, it proposes an alternative of the popular density based clustering algorithm DB-Scan, which modifies the proximity parameter (i.e. *epsilon*) of the algorithm. The proposed alternative employs in tandem the difference in (i) speed, (ii) course and (iii) position for defining the distance between two consecutive vessel positions (two consecutive AIS signals received from the same vessel). The results show that this combination performs significantly better than using only the spatial distance and, more importantly, results in clusters that have very interesting properties. The enriched network model can be processed and further examined with data mining techniques, even in an unsupervised manner, in order to identify anomalies in vessels' trajectories. Experimental results on a real dataset show the network's potential for detecting trajectory outliers and uncovering deviations on a vessel's route.

**Keywords:** Trajectory analytics · AIS vessel monitoring · Anomaly detection

# 1 Introduction

Today’s maritime surveillance systems are constantly flooded by data coming from AIS transponders, which are embedded in vessels. The use of AIS transponders was made compulsory for all vessels over 300 Gross Tonnage and all passenger vessels in 2002 by the Regulation 19 of SOLAS Chapter V<sup>1</sup>. However, even smaller vessels, from yachts to fishing boats [1], are now using AIS to report their positions to the nearby vessels, usually for safety purposes, making AIS the number one system for global vessel tracking. Each vessel transmits two kinds of AIS data, dynamic and static. The former, periodically sends data regarding vessel’s position, speed and heading. The transmission rate depends on the vessel’s speed and becomes higher when the speed is greater. The latter, sends data, every six minutes approximately, regarding vessel’s destination, type, size and draught of its hull.

Due to the fact that AIS data are sent periodically with high transmission rates, they are of utmost importance to the maritime authorities for vessel tracking purposes. Therefore, a system that takes advantage of such data and is able to notify the authorities in real-time for any abnormal vessel behavior can be valuable for the authorities. This work contributes directly towards anomaly detection from AIS data. It builds upon our previous work in the field [2], which defined a methodology for extracting a network abstraction of the maritime traffic in an area. The input in that work was a lengthy log of AIS data collected from vessels that sailed in that area and the output was a network representation model of the typical routes that the vessels have followed. In that network representation, the nodes (also called *way-points*) are regions of special interest for the routes of vessel and they usually correspond to ports, capes, offshore platforms etc., where multiple vessels usually stop for short or longer periods, or perform major changes in their direction. Similarly, the connections between nodes represent the vessel movement from one way-point to another and thus, a vessel trajectory is a traversal of the network, from a certain way-point to a distant way-point. This traversal either follows the existing connections (and the trajectory can be considered normal) or deviates and hops from one node to another, not directly connected, node. The aggregated information from all vessels that crossed a network connection are used to extract features for this connection (a potential sub-trajectory for other vessels), such as the average, minimum and maximum speed etc.

In this work, we take this simple aggregation one step ahead, and provide a methodology that can be used to process this multi-vessel information in a more proficient manner. The proposed method adds richer information to each connection that have been traversed by multiple vessels. To extend the previously proposed network abstraction we use a clustering algorithm, that manages to identify different movement patterns for the same connection. This information is then used as a reference in the analysis of a vessels’ journey and can allow to identify routes that deviate from the previously extracted patterns. Furthermore,

---

<sup>1</sup> <http://solasv.mcga.gov.uk/regulations/regulation19.htm>.

it builds upon the semantic information of the edges of the network abstraction and adds to these connections common patterns the vessels must follow in order to travel between two way-points. Therefore, the common pathways and behavior of the vessels in terms of space, speed and heading are integrated to the already proposed network abstraction.

The main idea behind this work is that vessels of the same type (e.g., cargo vessels) that travel towards the same destination, follow common routes that pass through certain way-points and have similar moving patterns such as the same speed or heading. The major contributions of this work are:

- A variation of the popular density based clustering algorithm (DBScan) that takes into account the difference in speed and course as well as the spatial distance of trajectory points and extracts common navigation behaviors.
- A framework for taking advantage of these common navigation behaviors, by constructing movement models for different regions and vessel types and using them to detect deviations from the models.

A framework like this, allows further analysis by using well-known network analysis or data mining techniques enabling easier understanding of the maritime traffic.

The rest of the paper is structured as follows. Section 2 summarizes the literature in the field of feature extraction from multiple trajectories and their use for trajectory comparison. It focuses on works that summarize historical data and build semantic models for an area. In Sect. 3, the proposed methodology of enriching the network abstraction model is presented in detail and Sect. 4 discusses the preliminary results of our methodology in anomaly detection. Finally, Sect. 5 concludes the paper by summarizing the presented methodology and highlighting the impact of this work in the domain of the maritime surveillance by showing the possible use cases in the field of anomaly detection.

## 2 Related Work

In the context of the proposed work, traffic network abstraction and anomaly detection is the main focus. As a network abstraction model, it is comparable to methodologies that compress or summarize trajectories from historical AIS data in order to improve maritime surveillance systems. As a methodology for anomaly detection it is comparable to techniques that use historical AIS data to detect abnormal or noteworthy patterns or events.

Several works on maritime surveillance have used grid partitioning of the surveillance area into tiles or hexagons [3] for mapping vessel trajectories to polylines or sequences of spatial indexes or key-points [4]. The proposed model is a more coarse-grained representation than other trajectory simplification methods that try to remove redundant AIS data, but still keeping a large amount of them. Such methods apply to single vessel trajectories, whereas the proposed method applies to multiple vessel trajectories in the same region. The proposed methodology results with a few key-points extracted from the set of trajectories – the

way-points – and a set of edges between them, that contain statistics extracted from the actual vessel trajectories, which are clustered by similarity. Section 3 shows that the edges connect way-points that are away from each other and edges contain sufficient information about the vessels’ journeys between each pair of way-points.

Many works the recent years try to build maritime traffic network representations from historical AIS data [5,6]. Arguedas et al. [5] propose a two-layer network: (i) an external layer that uses way-points as nodes/vertices and routes as edges/lines and (ii) an internal layer that consists of nodes or *breakpoints* that represent vessels’ changes in behavior and edges or *tracklets* that represent vessel trajectories. The former layer is a traffic network abstraction, while the latter is a network that provides information about each vessel layer individually. While an edge in the first layer can be a route from a port to another port, an edge in the internal layer comprises all the simplified trajectories (using Douglas-Peucker algorithm [7]) that sailed across this route.

The complexity of the internal layer raises scalability issues that can be seen in the analysis of a real dataset. It is characteristic that the use of the 454 complete port-to-port routes in the small area of the Baltic sea resulted in an internal layer with 2,095 tracklets. Our proposed model is similar to that of the external layer of [5] but provides a much richer internal layer, that maintains statistical information extracted from the trajectories of the sailing vessels. The resulting model significantly reduces the total amount of data contributed originally by the vessels, without losing its descriptive power.

Since maritime traffic networks are able to provide compressed information about vessel trajectories, their use seems to be essential for vessel motion analysis and abnormal behavior. The problem of anomaly detection in the maritime domain [8] has been the focus of research for many years, although in the recent years it started attracting more attention. From the early works on anomaly detection from Holst et al. [9] and the later works of Varlamis et al. [10] and Chatzikokolakis et al. [11] on the detection of search and rescue patterns, several representation models and algorithms have been developed to increase maritime situation awareness, identify potential illegal activities and detect anomalous patterns in the vessels’ trajectories.

In [18], Pallota et al. propose a methodology for anomaly detection through the use of a maritime traffic model. The model first extracts way-points or clusters from vessel positions or ports and creates or updates the properties of the vessels in the surveillance area. Way-points are extracted and route objects are created by clustering the extracted vessel flows, using the DB-Scan algorithm, which contain spatio-temporal and kinematic features. Probabilities are extracted to classify a set of vessel positioning observations to a route, then using the classified route a prediction is made for the future location. Finally, transition probabilities are used to detect if a vessel’s behavior deviates from normality. Authors in [12] compare two methodologies for anomaly detection which both use the Gaussian Mixture Model (GMM) with a different algorithm for clustering. The first one uses the Expectation Maximization (EM) algorithm while the

second one uses the greedy version of the EM algorithm. Both techniques consider momentary states of the vessel motion. As an extension of the approach proposed in [12], authors in [13] evaluate two models for detecting anomalies and their ability to distinguish simulated trajectories from real ones, the GMM and the Kernel Density Estimator (KDE). Results indicated that there is no significant difference in the performance of these two models.

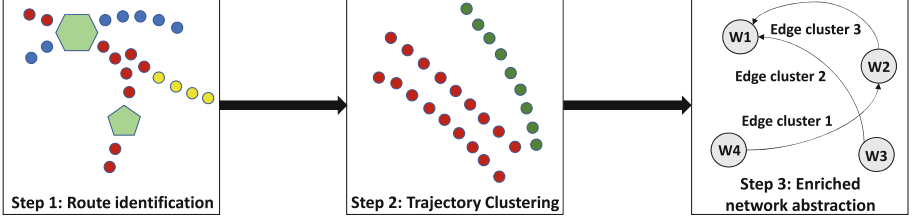
The proposed solution is expected to perform better than related frameworks for anomaly detection from AIS data, which employ the position information of the consecutive vessel signals that constitute its trajectory and use Euclidean or other distance metrics in a two-dimensional space (i.e., latitude and longitude) [14,15] or probabilistic approaches that partition space into tiles and estimate the probability of vessels to appear in a certain sequence of tiles [13] ignoring speed and direction. Even in approaches that use historical data to extract the average speed [16] or direction of move in a certain area [18], or techniques such as Piecewise Linear Segmentation (PLS) [17], speed and direction information are used only for predicting future vessel position and the detection of deviation always measures the spatial distance of the actual from the predicted position. From our knowledge, this is the first approach that builds a composite model of speed, direction and position for trajectories, which is then used to directly detect deviations of any of the three features or any combination of them. It is also expected to provide a richer model for the comparison of whole trajectories or sub-trajectories than the techniques that employ equal length sub-trajectories, or dynamic time warping and spatial distances to compare trajectories [19,20] or techniques that combine spatial and temporal dimensions for indexing trajectories [21].

### 3 The Proposed Approach

The proposed approach is applied to AIS data collected from multiple vessels of the same type (e.g., cargo vessels) for a predefined period of time and a predefined bounding box (e.g., geographic surveillance area of interest), but it is also applicable to larger geographic areas, periods of time and more types of vessels. Since, different types of vessels vary in size and shape, they may follow different routes even if they want to reach the same destination. Furthermore, specific vessel types such as cargo vessels might make much more intermediate stops (e.g. in middle sea platforms) than others. Although, the network abstraction model is the same for all types of vessels, the detailed information that it carries may vary per vessel type. So, in the following we present the model and the way its information is extracted but we demonstrate our approach on an AIS dataset from cargo vessels only.

The main steps of our approach are illustrated in Fig. 1.

- In the *route identification* step, the way-points are extracted from multi-vessel trajectory data, following a methodology proposed in [2] and summarized in Sect. 3.1. Vessel trajectories are then expressed as sequences of sub-trajectories that connect intermediate way-points.



**Fig. 1.** The steps of the proposed approach

- The (sub-)trajectory clustering step is the main contribution of this work, which introduces a novel use of the DB-Scan algorithm that takes into account 3 parameters to identify neighboring points. The methodology followed in this step is explained in details in Sect. 3.2.
- In the network abstraction model enrichment step, several statistics are extracted for each cluster. The statistics summarize the movement of multiple vessels along the network edge. The details of these statistics and their extraction method is given in Sect. 3.3.

The final output model, comprises a set of way-points (vertices) dispersed across the monitored region and several sub-trajectory clusters (edges) with their statistics per cluster to represent the different ways of moving between two way-points. This output can be used for many use cases in the field of anomaly detection.

### 3.1 Route Identification

The first step of our methodology is the identification of way-points, which represent areas where many vessels have stopped (stop points) or did a major directional change (turn points) in the past. As already demonstrated in [2], way-points are created by clustering stop and turn points using a spatial density clustering algorithm (i.e. DB-Scan). The resulting way-points are the nodes of the network abstraction model, which contains information about way-points' size and density (number of stop or turn points per area unit). The size and density of way-points is strongly connected to the parameters of the DB-Scan algorithm. In our working examples, we focus only on the bigger way-points (i.e. those that contain more than 50 points). The idea behind this filtering is that bigger and denser way-points would belong to the trajectories of more vessels.

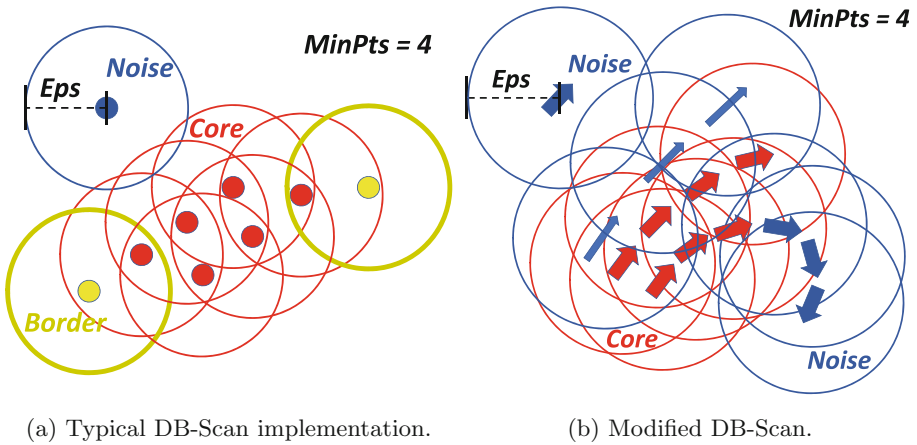
In our prototype analysis, we focus only on the trajectories that have at least 2 way-points, although the same methodology can be applied in all trajectories and respectively to all the edges of the network. Using different selection thresholds may result either in losing semantic information or in keeping too much information and this is a subject of further experimentation. For example, using higher thresholds (e.g. keeping even larger way-points only) will result in a higher level of abstraction and will probably loose the fine grained details of

multiple vessel patterns, whereas using lower thresholds will result in keeping too much information and achieve low or no abstraction at all.

### 3.2 Trajectory Clustering

The second step refers to the clustering of the trajectories that have the same origin and destination way-points. The typical algorithm for clustering the points of one or more trajectories is DB-Scan [22], which is employed as a density-based spatial clustering method. DB-Scan takes two parameters, *epsilon* which specifies how close two points must be to be considered neighbors, and *minPts* which specifies the number of neighbors a point must have to be included in a cluster. Our proposed DB-Scan version uses 3 parameters to specify the proximity of candidate vessel AIS signals (positions):

- *s*: absolute difference of the speed between two positions (speed-based)
- *h*: absolute difference of the course over ground between two positions (heading-based)
- *eps*: harvesine distance between two positions (spatial-based)



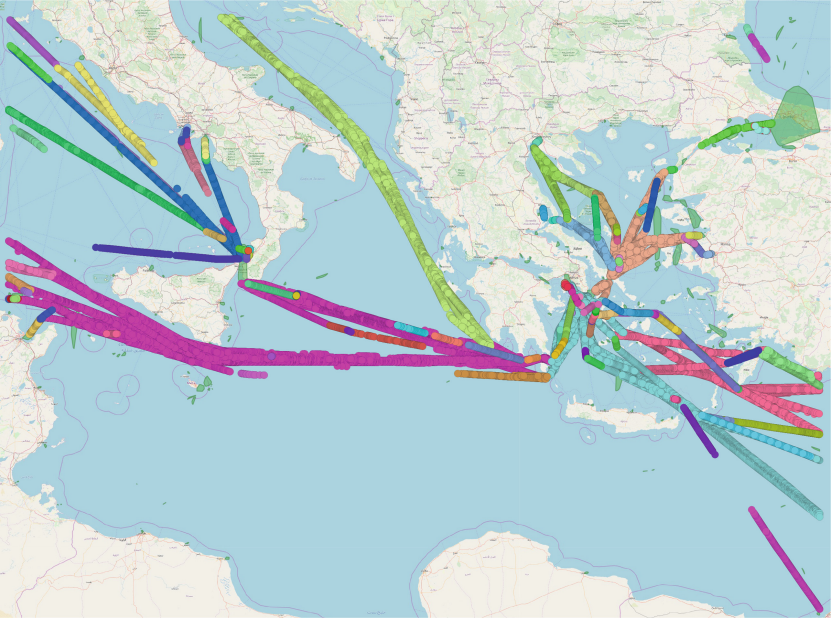
**Fig. 2.** Comparison of DB-Scan implementations.

To the best of our knowledge, this DB-Scan variation has not been used in the related literature. Therefore, each vessel position contains three types of information: (i) the vessel speed at this position, (ii) the vessel course over ground at this position, (iii) the latitude and longitude of the position. Also, for a vessel position to be clustered together with another vessel position, the absolute difference in their speed must be below a threshold *s*, the absolute difference in their heading below a threshold *h* and the distance between them must be below a threshold *eps* at the same time. This type of clustering groups



together trajectory points that have similar speed, heading and are close to each other. An example of this type of clustering can be seen in Fig. 2 which compares the two implementations of the DB-Scan algorithm. Figure 2a shows the typical DB-Scan implementation, which creates a cluster if points are spatially close to each other. On the other hand, Fig. 2b illustrates the modified DB-Scan for the positions of moving objects, which considers two points (actually two vectors with position, direction and speed) to be in the same neighborhood when the vectors' positions are spatially close to each other, but they also have similar direction and speed. In the modified version blue arrows indicate noise vectors, which are either away, or have different speed or have a different direction from all their neighbouring vectors.

To have more accurate clustering results, we exclude positions that are located inside the way-points. Since way-points are areas of interest through which vessels frequently pass, it can be easily inferred that the way-points might be ports, platforms, canals or waterways. Inside these way-points, vessels tend to alter their speed or heading frequently, which may corrupt the clustering results.



**Fig. 3.** Example of the trajectory clustering.

Figure 3 illustrates the result of running the proposed trajectory clustering method<sup>2</sup> to all cargo vessels that sail in the east Mediterranean sea and are headed to the port of Piraeus, Greece, using  $s = 3$ ,  $h = 3$ ,  $eps = 20$  km and

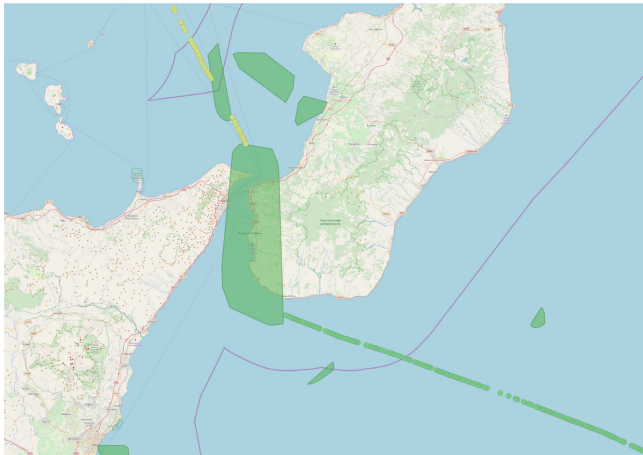
<sup>2</sup> DB-Scan parameters have been empirically determined.



$minPts = 10$ . We can see that trajectories with similar speed and heading are placed in the same cluster, which resembles to the behavior of the basic DBScan (e.g. the cluster formed in the Adriatic sea), whereas points of the same trajectory may belong to different clusters, even though they are spatially close, because of the differences in speed or heading (e.g. the clusters that are formed near the port of Tripolis, Lybia, on the left part of Fig. 3).

### 3.3 Enriched Network Abstraction

The final step of the process is the enrichment of the network model with information about the clusters of sub-trajectories in each network edge (or in selected edges, e.g. the most frequently traversed). Since, we have created clusters of trajectories (edges) between way-points, we can add information to these edges to form a comprehensive network of the maritime traffic. To this end, for each cluster or edge of the network we calculate the average travelling speed and heading of the vessels. Moreover, the typical deviation of these values is also calculated. Finally, the start point and the end point of the cluster are computed (beginning and ending of the trajectories) along with the average temporal distance of each cluster (average time taken to travel from the start of the cluster to the end of it). Figure 4 illustrates a small snapshot of the network near Sicily, Italy. The green shaded convex hulls represent way-points (vertices) and the green and yellow dots are the points that comprise the trajectory of a single vessel<sup>3</sup>. For the (yellow) subtrajectory points that connect the two way-points of the figure, the centroid of the respective cluster has a heading of 319.15 whereas the centroid of the other (green) (subtrajectory) has a heading of 322.28.



**Fig. 4.** Example of the edges of the network abstraction. (Color figure online)

<sup>3</sup> For demonstration purposes, this cluster contains points from a single vessel's trajectory.

## 4 Application to a Real Dataset

To examine the results of our enriched network model, a dataset provided by MarineTraffic<sup>4</sup> was used, containing 2.9 million AIS messages received from 1,716 distinct “cargo” vessels sailing in the eastern half of the Mediterranean Sea during August 2015. Since no information about the existence of anomalous behaviors existed in this dataset, we employed unsupervised techniques to detect potential anomalies or outliers. Although outliers can be detected, further examination is required to understand the reason behind the unusual behavior and the characteristics of the trajectories selected.

### 4.1 Network Creation from Real AIS Positions

The first step in building the enriched network abstraction is the creation of the way-points (vertices of the network). The identification of the way-points is a two-step process that requires to *i*) identify key-points in the trajectories of the vessels and *ii*) spatially cluster together dense key-points. To identify the key-points we used a speed threshold of 2 knots and a bearing rate threshold of 0.1 degrees per minute, which resulted in several thousand low speed AIS positions and turns in the trajectories of the surveillance area. To create the clusters of key-points, we used the DB-Scan algorithm with a minimum number of ten key-points ( $minPts = 10$ ) within a radius of 2 km ( $eps = 2000$ ), resulting in 616 clusters.

The second step involves clustering of the trajectories with similar characteristics. For this step, we grouped the trajectories per destination and applied the proposed modified version of DB-Scan, which requires for 3 parameters to be satisfied in order for a point to be considered a neighboring one (speed-based, heading-based, spatial-based). For a point to be in the same cluster, its speed must not differ more than 3 knots and its heading more than  $3^\circ$  within a 20 km radius. Moreover, a minimum number of 10 points is required to form a cluster.

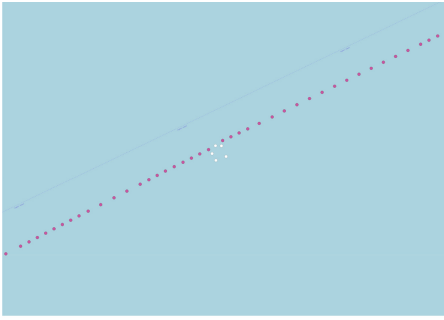
In the remainder of this section we demonstrate cases of vessels that had unusual behavior in terms of the way they deviate from their route or in terms of the way they suddenly change course to reach the same destination.

### 4.2 Detection of Outliers in the Trajectories

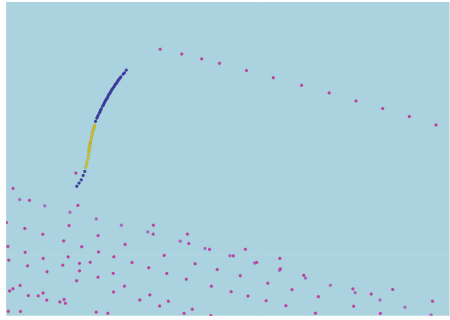
The lack of Maritime Situational Awareness (MSA) is a key factor in many incidents that are due to crew fatigue, stress or even engine failures, despite the major improvements in maritime safety. A sudden change in the course of a vessel is considered a noteworthy or anomalous event for the maritime authorities for several reasons, either due to human factors or technical ones. Several cases have been recorded in the past, in which engines fail during a vessel’s voyage and the vessel starts drifting away from its normal route. This type of deviation in a vessel’s route could potentially lead to collisions with nearby vessels

<sup>4</sup> <https://www.marinetraffic.com/>.

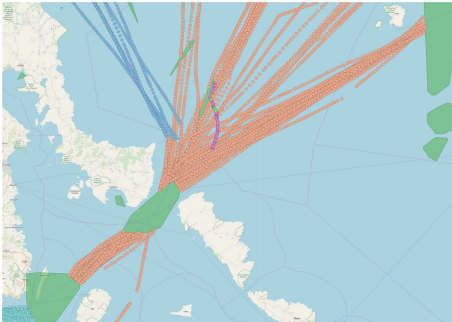
or collisions with rocky islands, endangering multiple vessels in the vicinity or the environment (e.g., oil spills). Such small deviations from the normal route cannot be detected by algorithms that seek for major turns, and the same holds for temporal decelerations or accelerations and algorithms that seek for sudden stops. Similarly, when vessels are in distress due to piracy attacks or when they take part in search and rescue operations and they perform manoeuvres, it is not always feasible to detect such combined actions that include speed and route change and deviation from the normal route. These types of behavior require an immediate course of action by the authorities. The proposed network abstraction model, with the information it carries on each edge concerning the clusters of movement patterns (in terms of speed, course over ground and location) is able to capture such cases that comprise small or larger deviations in the trajectories. A few outlier cases that have been detected (Fig. 5) on a real dataset are presented in the following.



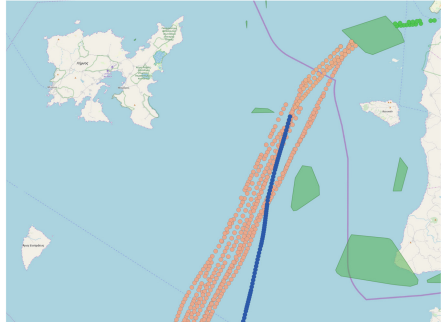
(a) Example of an unusual loop in a vessel's trajectory.



(b) Example of an unusual and steep deviation of a vessel.



(c) A trajectory that does not follow the usual maritime traffic has been detected.



(d) A trajectory that slowly deviates from its course.

**Fig. 5.** Outliers detected by the proposed trajectory clustering.

Figure 5a illustrates a vessel's trajectory towards Naples, Italy. During its voyage the vessel makes a small circle and then continues its journey as before. Since its heading and speed changed dramatically the points in the circle (i.e. white) are considered outliers. Figure 5b illustrates the maritime traffic from the west to east, near Sicily, Italy. The trajectories from multiple vessels are grouped in the same cluster, since they share the same course and speed values and are drawn with the same colour (i.e. magenta). The centroid of this cluster has a heading of  $102.3^\circ$  and a speed of 13.91 knots. However, the part of the trajectory of a vessel that deviates from the normal route, starts heading to the north and after a while follows the same direction as before is marked with blue and yellow dots, since it moved to a different cluster. The blue cluster centroid has a heading of  $30.2^\circ$  and a speed of 1.2 knots (with a standard deviation of 5.25), whereas the respective centroid for the yellow cluster has a heading of  $11.4^\circ$  and a speed of 1.2 knots (with a standard deviation of 2.75). The actual centroid values clearly indicate an outlying behavior from a vessel that changed its route in slow speed in an area where similar (i.e. cargo) vessels move in different speed and direction. In a different case, Fig. 5c visualizes the maritime traffic of cargo vessels in the Aegean sea, showing all the vessels heading to the port of Piraeus, passing south of the island of Evia and near the island of Andros, Greece. There are two distinct clusters in the plot: (i) a big one that contains the trajectory of vessels traveling from the north-east Aegean sea, with a centroid of  $227.3^\circ$  (stdev = 21.32) and 13.1 knots (stdev = 2.49) and (ii) one that contains vessels traveling from the north-west, with a centroid of  $137.8^\circ$  (stdev = 14.26) and 13.0 knots (stdev = 1.65). The two clusters eventually merge into one cluster when the vessels pass south of Evia. Almost hidden among the two clusters is a third smaller cluster (marked with purple points) which illustrates a large deviation of a vessel that does not follow the patterns of all vessels with the same destination. This last cluster has a centroid of  $145.8^\circ$  (stdev = 2) and 1.4 knots (stdev = 0.16). With the proposed clustering algorithm, this subtrajectory, which does not contain any large and sudden course change or a stop has been identified as an outlier. Finally, Fig. 5d shows the maritime traffic near the island of Lemnos, Greece. From the plot it is obvious that while all vessels follow a specific route (the same big cluster as in Fig. 5c), when they head towards the port of Piraeus, using similar speed and heading values, there is one vessel that slowly deviates (marked with blue colour) from the common route, for unknown reason. This outlier has an average heading of  $192.7^\circ$  and 9.8 knots speed. The comparison between the normal behavior ( $227.3^\circ$ , with stdev = 21.32 and 13.1 knots, with stdev = 2.49) shows that this outlier moved much slower than all other cargo vessels too.

All the cases presented above, are extracted from a dataset of 1,716 cargo vessels, following a totally unsupervised method (clustering). As a consequence, it provided us with useful feedback on the applicability of the proposed method and on the type of deviations it can detect. However, the same methodology can be used as a basis for a supervised (classification) technique that will detect vessel deviations using pretrained cluster information.

## 5 Conclusion and Future Steps

In this work, we proposed a clustering technique, which can be used to enrich our previously proposed maritime traffic network [2] that can efficiently model the behavior of vessels using only free and openly transmitted AIS data. The modelling of the normal vessel behavior will allow us to further distinguish outliers in the trajectories that are of interest to the maritime authorities. In this work, we showcased a few real world examples which our model managed to accurately detect. Identifying specific cases of anomalous behavior [10, 11, 23, 24] will allow us to fine-tune, improve and exploit the proposed unsupervised technique as a basis for a supervised model for the detection of events of interest in the maritime sector. As a future work, we intend to exploit the proposed network abstraction in order to identify events of interest to the maritime authorities. Besides the route deviation problem presented in the preliminary results, we are interested in identifying several other anomalies related to the maritime domain such as communication gaps, AIS spoofing and illegal activities, thus building a unified framework for anomaly detection in real-time. The evaluation of the future anomaly detection framework will take into account real-world incidents and will measure the detection latency in real-time.

**Acknowledgements.** This work has been developed in the frame of the MASTER and SmartShip projects, which have received funding from the European Union’s Horizon 2020 research and innovation programme under the Marie Skłodowska-Curie grant agreement No. 777695 and No. 823916 respectively.

## References

1. Montewka, J., Kujala, P., Ylitalo, J.: The quantitative assessment of marine traffic safety in the Gulf of Finland, on the basis of AIS data. *Zeszyty Naukowe/Akademia Morska w Szczecinie*, pp. 105–115 (2009)
2. Varlamis, I., Tserpes, K., Etemad, M., Júnior, A.S., Matwin, S.: A network abstraction of multi-vessel trajectory data for detecting anomalies. In: *Proceedings of the Workshops of the EDBT/ICDT 2019 Joint Conference*, Portugal, Lisbon, March 2019
3. Yap, P.: Grid-based path-finding. In: Cohen, R., Spencer, B. (eds.) *AI 2002. LNCS (LNAI)*, vol. 2338, pp. 44–55. Springer, Heidelberg (2002). [https://doi.org/10.1007/3-540-47922-8\\_4](https://doi.org/10.1007/3-540-47922-8_4)
4. (PDF) mR-V: Line Simplification through Mnemonic Rasterization
5. Fernandez Arguedas, V., Pallotta, G., Vespe, M.: Maritime traffic networks: from historical positioning data to unsupervised maritime traffic monitoring. *IEEE Trans. Intell. Transp. Syst.* **19**(3), 722–732 (2018)
6. Coscia, P., Braca, P., Millefiori, L.M., Palmieri, F.A.N., Willett, P.: Multiple ornstein-uhlenbeck processes for maritime traffic graph representation. *IEEE Trans. Aerosp. Electron. Syst.* **54**(5), 2158–2170 (2018)
7. Douglas, D.H., Peucker, T.K.: Algorithms for the reduction of the number of points required to represent a digitized line or its caricature. *Cartogr. Int. J. Geogr. Inf. Geovis.* **10**(2), 112–122 (1973)

8. Holst, A., Ekman, J.: Anomaly detection in vessel motion (2003)
9. Holst, A., et al.: A joint statistical and symbolic anomaly detection system: increasing performance in maritime surveillance. In: 2012 15th International Conference on Information Fusion, pp. 1919–1926, July 2012
10. Varlamis, I., Tserpes, K., Sardianos, C.: Detecting search and rescue missions from AIS data. In: 2018 IEEE 34th International Conference on Data Engineering Workshops (ICDEW), pp. 60–65, April 2018
11. Chatzikokolakis, K., Zissis, D., Spiliopoulos, G., Tserpes, K.: Mining vessel trajectory data for patterns of search and rescue. In: Proceedings of the Workshops of the EDBT/ICDT 2018 Joint Conference (EDBT/ICDT 2018), Vienna, Austria, 26 March 2018, pp. 117–124 (2018)
12. Laxhammar, R.: Anomaly detection for sea surveillance. In: 2008 11th International Conference on Information Fusion, pp. 1–8, June 2008
13. Laxhammar, R., Falkman, G., Sviestins, E.: Anomaly detection in sea traffic - a comparison of the Gaussian mixture model and the kernel density estimator. In: 2009 12th International Conference on Information Fusion, pp. 756–763, July 2009
14. Fu, Z., Hu, W., Tan, T.: Similarity based vehicle trajectory clustering and anomaly detection. In: IEEE International Conference on Image Processing 2005, vol. 2, pp. II–602. IEEE (2005)
15. Hexeberg, S., Flåten, A.L., Brekke, E.F., et al.: AIS-based vessel trajectory prediction. In: 2017 20th International Conference on Information Fusion (Fusion), pp. 1–8. IEEE (2017)
16. Etemad, M., Soares Júnior, A., Matwin, S.: Predicting transportation modes of GPS trajectories using feature engineering and noise removal. In: Bagheri, E., Cheung, J.C.K. (eds.) Canadian AI 2018. LNCS (LNAI), vol. 10832, pp. 259–264. Springer, Cham (2018). [https://doi.org/10.1007/978-3-319-89656-4\\_24](https://doi.org/10.1007/978-3-319-89656-4_24)
17. Le Guillarme, N., Lerouvreur, X.: Unsupervised extraction of knowledge from S-AIS data for maritime situational awareness. In: Proceedings of the 16th International Conference on Information Fusion, pp. 2025–2032. IEEE (2013)
18. Pallotta, G., Vespe, M., Bryan, K.: Vessel pattern knowledge discovery from AIS data: a framework for anomaly detection and route prediction. *Entropy* **15**(6), 2218–2245 (2013)
19. Lee, J.-G., Han, J., Whang, K.-Y.: Trajectory clustering: a partition-and-group framework. In: Proceedings of the 2007 ACM SIGMOD International Conference on Management of Data, pp. 593–604. ACM (2007)
20. Etemad, M., Júnior, A.S., Hoseyni, A., Rose, J., Matwin, S.: A trajectory segmentation algorithm based on interpolation-based change detection strategies. In: Proceedings of the Workshops of the EDBT/ICDT 2019 Joint Conference, Portugal, Lisbon, March 2019
21. Tampakis, P., Pelekis, N., Andrienko, N., Andrienko, G., Fuchs, G., Theodoridis, Y.: Time-aware sub-trajectory clustering in Hermes@ PostgreSQL. In: 2018 IEEE 34th International Conference on Data Engineering (ICDE), pp. 1581–1584. IEEE (2018)
22. Ester, M., Kriegel, H.-P., Sander, J., Xu, X.: A density-based algorithm for discovering clusters a density-based algorithm for discovering clusters in large spatial databases with noise. In: Proceedings of the Second International Conference on Knowledge Discovery and Data Mining, KDD 1996, pp. 226–231. AAAI Press (1996)

23. Kontopoulos, I., Spiliopoulos, G., Zissis, D., Chatzikokolakis, K., Artikis, A.: Countering real-time stream poisoning: an architecture for detecting vessel spoofing in streams of AIS data. In: IEEE joint conferences DASC/PiCom/DataCom/CyberSciTech 2018, pp. 981–986, August 2018
24. Patroumpas, K., Alevizos, E., Artikis, A., Voudas, M., Pelekis, N., Theodoridis, Y.: Online event recognition from moving vessel trajectories. *Geoinformatica* **21**(2), 389–427 (2017)

**Open Access** This chapter is licensed under the terms of the Creative Commons Attribution 4.0 International License (<http://creativecommons.org/licenses/by/4.0/>), which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license and indicate if changes were made.

The images or other third party material in this chapter are included in the chapter's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the chapter's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder.

