sRADAR: A Complex Event Processing and Visual Analytics System for Maritime Intelligence

Naveen Nandan **Baljeet Malhotra*** **Daniel Dahlmeier**

SAP Research and Innovation, CREATE #14
University Town, 1 Create Way, Singapore 138602

naveen.nandan@sap.com baljeet.malhotra@sap.com d.dahlmeier@sap.com

Abstract

Maritime Intelligence is about empowering users in a port ecosystem with data and visual analytics to increase the efficiency and effectiveness of maritime operations. In this context, discovery and visualization of *ship domain violations* based on the analysis of trajectories generated by ships could serve important navigational and business purposes. Finding patterns of domain violations in a large trajectory database, however, is a non-trivial task, primarily due to the combinatorial nature of the problem. In this paper, we present a system, sRADAR, which models such trajectories, applies complex event processing on the data streams to identify such domain violations, performs analytics to derive useful insights into the recorded data, and helps visualizing the result of such geo-spatial analytics. To evaluate our proposal, we setup an Automatic Identification System for collecting real trajectories of ships arriving at the port of Singapore. We discuss some preliminary results on domain violations and the efficiency of our system using the real-time data collected by our system.

Keywords: Anomaly detection, Complex event processing, Maritime Intelligence, Visual analytics.

1 Introduction

Interactions between ships are an important area of research in marine navigation science and traffic engineering. According to the International Maritime Organization, 90% of the global trade is transported by sea, and as the global trade is increasing, ship collision avoidance will become more important than ever before. In this context, *ship safety domain* is a key concept that essentially prescribes an area around a ship that must not be intruded by any other ships for safe navigation. Ship safety domain is very critical, not only to enhance the navigation safety, but also to protect the lives of crew members and the serenity of marine environment. When a ship enters into the safety domain of another ship, it is generally called *ship brushing* or *domain violation*. Apart from jeopardizing human and financial losses, ship brushing may have serious implications on international relations.

*Corresponding Author.

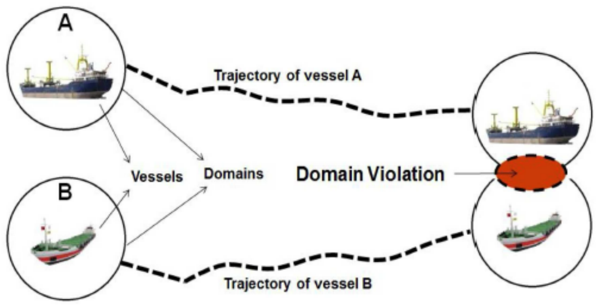

Figure 1: An illustration of a domain violation between two vessels (ships) named A and B.

A scenario of ship domain violation is depicted in Figure 1. The severity of ship brushing depends on the speed, size, and the orientation of the ships involved. Though domain violations are unavoidable in some situations, e.g., in narrow water channels where space is limited and may not always lead to accidents, domain violations can be serious and serve as an indicator of navigation safety. For instance, one can ask the following questions which are not only important for port authorities, but also for insurance companies and the vessel owners:

1. What types of vessels cause more violations than others?
2. In which geographical regions (that are bottlenecks) are violations more frequent?
3. Are there any particular seasons when the violations occur more frequent than others?

Answers to the above questions could be useful for situational awareness, port capacity planning, vessel profiling, and insurance claims.

In this paper, we describe sRADAR, a system that aims at automatically identifying complex events related to such questions and alerting various stake holders, such as port authorities, port operators, shipping companies, and insurances, of such domain violations. In order to provide further insight into the detected events, we perform complex data analysis on the collected geo-spatial and temporal data. Visualization and interaction can bridge the gap between computational data analysis methods, human reasoning, and decision-making processes, combining the strengths of both worlds (Riveiro 2011, Riveiro & Falkman 2011). On one hand, we take advantage of intelligent algorithms and the vast computational power of modern technology, such as in-memory databases, and on the other hand, we integrate human ability to comprehend information using intuitive methods of visualization for the derived knowledge. To track the position of ships, we rely on the Automatic Identification System (IMO 2001) or AIS for short, which provides a rich source of data on ships' identification, trajectories, navigation status and others. The AIS technology is explained in more detail in Section 3.

The rest of the paper is organized as follows. Section 2 presents a discussion on the related work in the context of trajectory data mining and analytics. In section 3, we present an overview of the system and the details of our experimental setup. In the following section 4, we discuss the architecture and other technical details of the sRADAR system. Section 5 presents the details of the analytics visualization and reporting component of the system. The paper is concluded in section 6.

This paper discusses the preliminary approach taken towards building a solution for the maritime scenario. In building this prototype, we use and exploit some of the capabilities of in-memory database technology and front-end technology, HANA© and UI5© respectively, offered by SAP.

2 Related Work

Many researchers have extensively used maritime data specifically generated by the AIS for trajectory data mining (Li et al. 2010). The AIS data has also been used to study a spectrum of multidisciplinary problems such as maritime emission control (Perez et al. 2009), anomaly detection and risk assessments (Ristic et al. 2008, Laxhammar et al. 2009, Jakob et al. 2010), complex network analysis (Kaluza & et. al. 2010), and others (Malhotra et al. 2011).

Trajectory data mining is an emerging and rapidly developing topic in the area of data mining that aims at discovering patterns of trajectories based on their proximity in either a spatial or a spatio-temporal sense. As ships keep moving and continuously generate trajectory data, mining their trajectories plays an important role in maritime data management (Alvares et al. 2007, Andrienko et al. 2007, de Vries et al. 2010, Giannotti et al. 2007, Lee et al. 2008, Li et al. 2010). For instance, at a commercial port where hundreds of vessels may enter or leave the port or wait to do so, collision avoidance is of utmost importance (Statheros et al. 2008).

Trajectory data mining methods can also be employed to discover mobility, traffic and congestion patterns which can then be used for situational awareness (Alvares et al. 2007). Based on the movement patterns, trajectories (and the vessels spanning them) can be clustered into groups to access the interactions between them and their collision risks (Li et al. 2010, de Vries & van Someren 2008). Furthermore, models can be built based on the discovered patterns to engineer monitoring systems such as the one proposed in (Piciarelli & Foresti 2006), which can then be used to detect anomalies (e.g., the trajectory of a particular ship that is not adhering to the guidelines) in real time to warn the authorities immediately. In (Perez et al. 2009), the authors discussed the challenges of data management, analysis, and the problems of missing data in the AIS datasets while proposing potential methods for addressing the limitations. Yet another study (Malhotra et al. 2011), discusses the management of the AIS data streams from the perspective of privacy and access control.

The purpose of this paper is not to conduct an in-depth survey of works that deal with the above data mining techniques or to focus on trajectory analysis based on the AIS data. Rather we pay attention to the particular problem of domain violations that could serve various purposes. We also focus on visualization and reporting mechanisms that are important for analyzing the brushing incidents (in particular in port waters) for various stake holders of a port ecosystem as mentioned previously.

Field	**Description**
MMSI	Mobile Marine Service Identifier. 9 digit identifier for a vessel's AIS.
Navigation Status	For example, *under way using engine*, *at anchor*, *engaged in fishing*.
Rate of Turn	Turning rate in degrees per minute.
Speed	Speed of vessel in knots.
Longitude	Longitude position of vessel.
Latitude	Latitude position of vessel.

Table 1: Data fields in AIS message types 1 to 3.

3 AIS Overview

The Automatic Identification System (AIS) (IMO 2001)) is an automatic identification and tracking system for maritime vessels. The primary purpose of AIS is to improve navigation safety and avoid collisions between vessels. It allows ships and stations to broadcast messages that contain information about a ship's navigational status, position course and speed, among many others. The primary purpose for AIS is navigation safety and traffic control. The International Maritime Organization (IMO) requires all international voyaging ships with a gross tonnage of 300 or more tons and passenger ships to have an Automatic Identification System (AIS) installed.

AIS messages are broadcasted through VHS radio equipment. AIS messages can be received by other ships and by vessel traffic services stations in the vicinity (typically within 40 nautical miles). As a matter of fact, AIS messages can be received by anyone in range using an appropriate receiver and decoding hardware and software. The AIS protocol specifies 27 different message types which carry different information, for example ship navigation information, base station reports, information about ship size and dimensions, and search-and-rescue aircraft reports. In practice, we observe that AIS message types 1 to 3, which are position reports for navigation purposes, are used more frequently than the other message types. Table 1 shows some of the important fields contained in AIS message types 1 to 3. It is worth noting that AIS does not infer the position or navigational status of the ship automatically. AIS is merely a radio technology for broadcasting a ship's information, which has to be provided by other sensors on-board that ship, for example its GPS device.

3.1 Experimental Setup

The AIS data can be collected from an AIS communication network while using a multi-layer system typically consisting of a Complex Event (Stream) Processing Engine (CEP) (Arasu et al. 2003, Abadi et al. 2003) and a database system for processing, storing

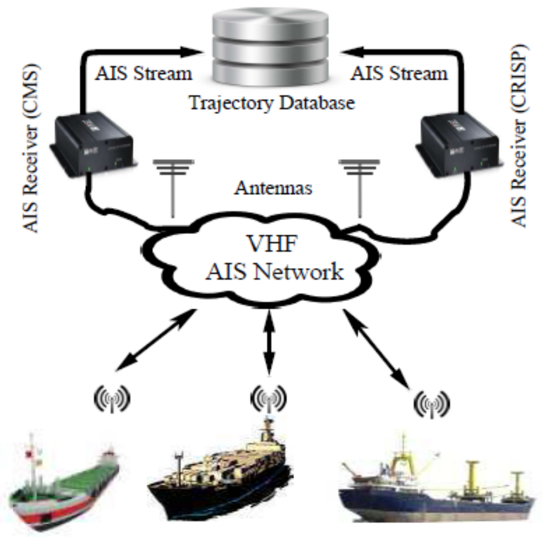

Figure 2: Experimental setup for data collection.

and analysis of the AIS data. At SAP Research, we have setup an AIS station to collect data from the ships arriving at the port of Singapore. The overall infrastructure of the setup is shown in Figure 2.

The captured AIS data is being processed and analyzed using a specialized complex processing engine (CEP). CEPs usually do not store data permanently, however, they allow access to traditional databases such as Oracle© and SQL Server© for data storage and processing purposes. We interfaced our CEP with HANA©, which is SAP's in-memory database appliance. Next, we describe the overall system architecture of the sRADAR system that we built for complex event processing based on AIS data and visual analytics for the purpose of maritime intelligence.

4 sRADAR Architecture

The proposed system consists of the following four main components:

1. AIS decoder,
2. rule engine,
3. real-time database,
4. visualization and reports.

Figure 3 shows the overall architecture of the system. We describe each component in turn.

4.1 AIS Decoder

The AIS messages are encoded in a binary format and hence, need to be decoded for further processing to identify information about the ships and their location. For this, we stream the AIS messages to an AIS decoder on the application server via UDP. The AIS decoder acts as a listener and as each message arrives, applies a decoding algorithm and stores the information to the database for future analysis. The decoder also forwards the ship information to the rule engine in order to perform on-the-fly computations, such as event detection.

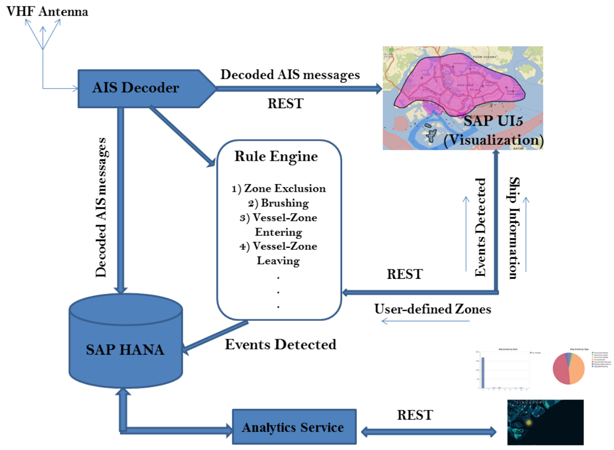

Figure 3: Overall architecture of the system.

4.2 Rule Engine

The rule engine has predefined rules, such as vessel brushing detection, vessel zone entering, vessel zone leaving, etc. As the decoded AIS messages arrive at the rule engine, each of the rules is applied to every message and if it matches the defined constraint, an event is fired that contains the relevant information about the detected event, for example the location, the identifier of the ships involved, and the time. The detected events are stored in the database which is used for further analytics as well as reporting and visualization.

4.3 Real-Time Database

The database layer for the system is implemented in SAP HANA© which is highly effective for temporal analysis of geo-spatial data. The database acts as both data store for the event detection phase and as the analytical engine for complex queries which are described in the following section. The initial system prototype makes use of JDBC calls to interact with the database, but going forward we are investigating on exploiting the geo-spatial capabilities of HANA©.

4.4 Visualization and Reports

The visualization layers for the prototype are built using a mix of SAP UI5© for the real-time ship movements, event reports and drawing zones for geo-fencing, CVOM©, bundled within UI5©, for analytical charts and Leaflet© for the event heatmap and derived event trajectory visualization. We considered visualization to be one of the most important components of the sRADAR system as visual analytics are not only important for reporting incidents, but also for situational awareness. To that end, we discuss the Visual Analytics in detail in the following section.

5 Visual Analytics

The system allows users to visualize data in different modes.

1. Real-time Ship Movement
2. Event Reports
3. Analytical Charts
4. Event Heatmap

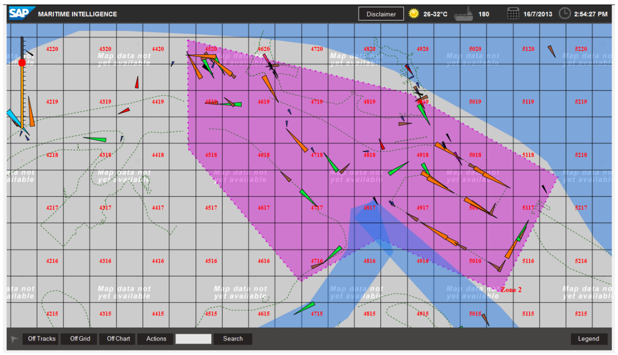

Figure 4: Real-time visualization of ships and interface to perform various actions such as drawing of zones.

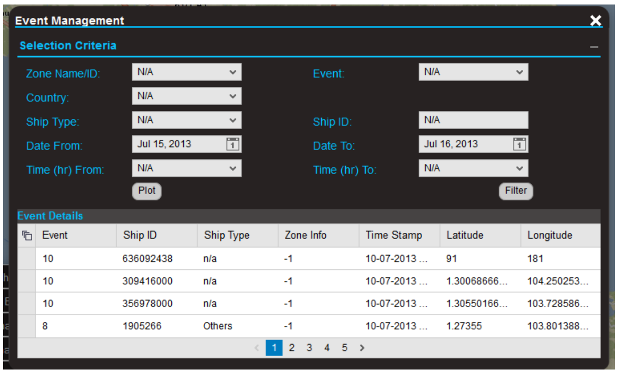

Figure 5: User interface for the selection of detected events based on various criteria such as ship/zone types, timeline, country of origin and so on.

5. Event Spatial Clustering
6. Brushing Event Trajectory

5.1 Real-time Ship Movement

The decoded AIS messages contain information about the position of ships. As a new AIS message arrives, the system identifies the source and checks if that specific ship has been identified before. If the ship has not been detected before, we store the information in the database along with its latest position. In case the ship already exists in the database, a comparison is made with its previous reported position and if this has changed, the latest position is updated. In both cases, the updated position is returned to the frontend which in turn plots the ship on a map. The mapping API used for this real-time plot is provided by ESRI©. Apart from plotting objects on the map, the front-end can also be used to draw and define zones for geo-fencing. This is illustrated in Figure 4.

5.2 Event Reports

The events triggered are stored in the database with information on time of occurrence, event type, position of the ship, details of other ships involved in the event, zone information, if any, and so on. These are presented to the user in the form of tabulated reports. This service is available on demand i.e., the query is triggered when user requests through the UI and populated in tabular format accordingly. An example of the event report UI is shown in Figure 5.

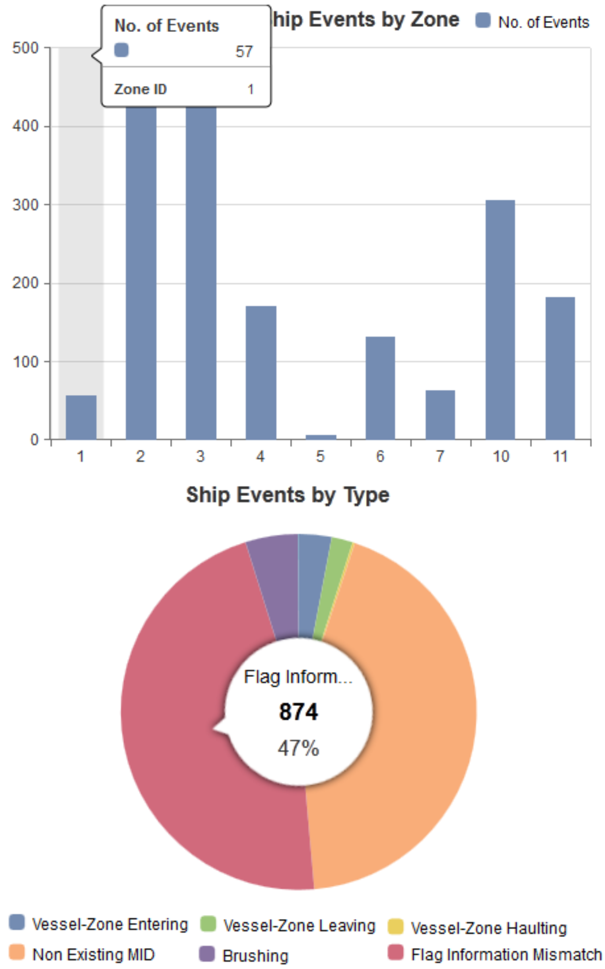

Figure 6: Analytical charts that present ship events by zone and event types.

5.3 Analytical Charts

As the decoded AIS data grows in volume, the amount of events detected also increase proportionally. This makes it difficult to query for all the events and represent them in a report format. For this reason, we make use of the interactive charting library CVOM© to present such information in a more intuitive fashion. The queries that populate the charts aggregate events by ship type, event type, etc. and help the users to get a broader insight into the recorded data. This would in turn help the user to identify outliers and move into investigating further based on the knowledge gained.

The frontend makes use of REST calls to the backend that fire the query every few seconds, thereby, ensuring real-time updates. The data is populated on the charts only if there is a change in state of the result, making the process of calculation and retrieval quite efficient. Figure 6 shows examples of analytical charts from our system.

5.4 Event Heatmap

When a large number of data points are involved that are geo-spatially distributed, plotting each of them was found to be inefficient. After investigating further, using heatmaps was found to be a common method of presenting the density of an aggregate of spatially distributed points. We make use of the heatmap.js javascript library which is used as

Figure 7: Visualization of a heatmap based on detected events.

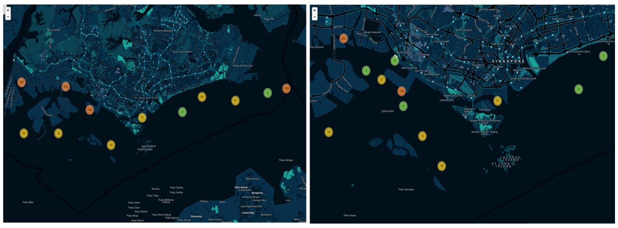

Figure 8: Visualization of spatially clustered events based on various zoom levels.

a layer atop leaflet.js, a mapping solution that integrates with OpenStreet© maps.

The user is allowed to select a specific event type, on which a REST call is made to query the database and retrieve a distribution of all points in space of that specific event. The returned points are plotted as a heatmap with the color intensity depicting the density of points returned. An example heatmap is shown in Figure 7.

5.5 Event Spatial Clustering

Another method to visualize a large number of spatially distributed points was to make use of the marker clustering API of Leaflet. By using this, we are able to cluster the number of events and display them as a collective point with an aggregated count, which in turn is controlled by the zoom level of the map. When the user zooms in, the clusters breakdown and render into individual points or smaller clusters as shown in Figure 8.

5.6 Brushing Event Trajectory

Brushing between ships was identified to be one of the major concerns at busy ports such as Singapore. We define a prior rule to detect brushing events between ships which is triggered when two ships violate the space constraint. The constraint is usually defined as the minimum distance that is to be maintained between any two ships. This minimum distance varies from ship to ship and takes into account the type and dimensions of the ship. An event is triggered and stored in the database for violation of such constraints with information on the ships involved, the time when the event occurs, the information on the zone in which the event occurred, if any, and the position of the ship. For every brushing event, two records are generated, one for each ship. Also, based on the duration of brushing there could be multiple records of the same event that is being recorded with different timestamp and position logs. Initially we present to the user a report of all such logs in tabular format, which was useful when the number of brushing events detected were quite low. But as more and more of such events were detected, the number of such records in the database grew large and representing them in tabular format was not always useful. For instance,

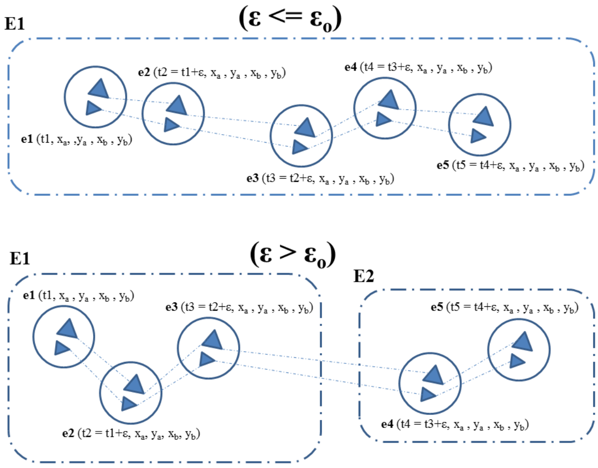

Figure 9: A scenario depicting multiple brushing events and a method to aggregate them based on time windows.

if our system detects brushing events between ship A and ship B for a period of 1 hour, based on the frequency of AIS messages received from these ships, we would have multiple entries of the same event with a variation in timestamp and position of the ships. Although this was accurate and represented the actual situation, from the user's perspective, it would not be useful to know that this event occurred multiple times.

To tackle this problem, and present the information to the user in a more intuitive form, we developed an algorithm to aggregate such events based on a time interval. A query is generated to aggregate all events of this type for each ship based on a time window. The definition of the interval for the time window is set based on studying the frequency of such events. Figure 9 represents the definition of the threshold between consecutive events. Here, *e1, e2, e3, e4, e5* are the event records as detected by the rule engine and stored in the database. The algorithm aggregates these events as a single brushing incident E1 or as two different brushing incidents *E1* and *E2* based on the variation of the threshold ϵ_0. In this way, multiple entries of the same brushing event that are recorded can be reported to the user as an aggregated event. One of the methods to present this information to the user is as before, using tabulated reports with the following attributes: *Ship1, Ship2, Start_Time, End_Time, Brush_Duration, Start_Position, End_Position.* Another method is to directly visualize the trajectory of the incident on a map. This is done by returning the result in GeoJSON standard format to the frontend Leaflet layer.

6 Conclusion

The system developed serves as a research prototype for maritime intelligence using AIS broadcast messages recorded from the ships around the port of Singapore, one of the busiest ports in the world. The capabilities of technologies such as in-memory databases can be further exploited for analytics on such large data volumes to detect anomalies in maritime traffic. In the future, the system will run analytics on multiple other data sources that provide accurate position information such as radar, surveillance cameras, etc. As the system for collecting and analyzing AIS messages is in place, further research focus would be in the direction of applying and developing learning algorithms to detect domain violations automatically, rather than handcrafting them. Also, the developed prototype is currently being evaluated by various stakeholders and further changes would be made based on the feedback.

7 Acknowledgment

We would like to thank the Economic Development Board of Singapore and National Research Foundation of Singapore for partially supporting this research. Our thanks are also due to SAP COIL and SAP HANA Solutions teams in Singapore for providing infrastructure support.

References

Abadi, D. J., Carney, D., Çetintemel, U., Cherniack, M., Convey, C., Lee, S., Stonebraker, M., Tatbul, N. & Zdonik, S. B. (2003), 'Aurora: a new model and architecture for data stream management', *The VLDB Journal* **12**(2), 120–139.

Alvares, L. O., Bogorny, V., de Macedo, J. A. F., Moelans, B. & Spaccapietra, S. (2007), Dynamic modeling of trajectory patterns using data mining and reverse engineering, *in* 'Proc. of the Int. Conf. on Conceptual Modeling (ER2007)', pp. 149–154.

Andrienko, G., Andrienko, N. & Wrobel, S. (2007), 'Visual analytics tools for analysis of movement data', *ACM SIGKDD Explorations* **9**(2), 38–46.

Arasu, A., Babcock, B., Babu, S., Datar, M., Ito, K., Nishizawa, I., Rosenstein, J. & Widom, J. (2003), Stream: The stanford stream data manager, *in* 'Proc. of SIGMOD'.

de Vries, G. K. D., van Hage, W. R. & van Someren, M. (2010), Comparing vessel trajectories using geographical domain knowledge and alignments, *in* 'Proc. of the ICDM Int. Workshop on Spatial and Spatiotemporal Data Mining (SSTDM)', pp. 209–216.

de Vries, G. & van Someren, M. (2008), Unsupervised ship trajectory modeling and prediction using compression and clustering, *in* 'Proc. of the Belgian-Netherlands Conf. on Artificial Intelligence', pp. 7–12.

Giannotti, F., Nanni, M., Pinelli, F. & Pedreschi, D. (2007), Trajectory pattern mining, *in* 'Proc. of the ACM Int. Conf. on Knowledge Discovery and Data Mining (KDD)', pp. 330–339.

IMO (2001), Guidelines for the onboard operational use of shipborne automatic identification systems (ais), as amended by itu-1371. resolution a.917(22), Technical report, International Maritime Organization.

Jakob, M., Vanek, O., Urban, S., Benda, P. & Pechoucek, M. (2010), Employing agents to improve the security of international maritime transport, *in* 'Proc. of the Int. Conf. on Autonomous Agents and Multiagent Systems', pp. 29–38.

Kaluza, P. & et. al. (2010), 'The complex network of global cargo ship movements', *The Journal of the Royal Society Interface* **7**, 1093–1103.

Laxhammar, R., Falkman, G. & Sviestins, E. (2009), Anomaly detection in sea traffic - a comparison of the gaussian mixture model and the kernel density estimator, *in* 'Proc. of the 12th Int. Conf. on Information Fusion', pp. 756–763.

Lee, J., Han, J. & Li, X. (2008), Trajectory outlier detection: A partition-and-detect framework, *in* 'Proc. of the Int. Conf. on Data Engineering (ICDE)', pp. 140–149.

Li, Z., Lee, J.-G., Li, X. & Han, J. (2010), Incremental clustering for trajectories, *in* 'Proc. of Int. Conf. on Database Systems for Advanced Applications', pp. 32–46.

Malhotra, B., Tan, W.-J., Cao, J., Kister, T., Bressan, S. & Tan, K.-L. (2011), Assist: Access controlled ship identification streams, *in* 'Proc. of the 19th ACM SIGSPATIAL Int. Symposium on Advances in Geographic Information Systems (GIS)', pp. 485–488.

Perez, H., Chang, R. & Billings, R. (2009), Automatic identification systems (AIS) data use in marine vessel emission estimation, *in* 'Proc. of the 18th Annual Int. Emission Inventory Conference'.

Piciarelli, C. & Foresti, G. L. (2006), 'On-line trajectory clustering for anomalous events detection', *Pattern Recognition Letters* pp. 1835–1842.

Ristic, B., Scala, B. L., Morelande, M. & Gordon, N. (2008), Statistical analysis of motion patterns in AIS data : Anomaly detection and motion prediction, *in* 'Proc. of the 11th Int. Conf. on Information Fusion', pp. 40–46.

Riveiro, M. & Falkman, G. (2011), The role of visualization and interaction in maritime anomaly detection, *in* 'IS&T/SPIE Electronic Imaging', International Society for Optics and Photonics, pp. 78680M–78680M.

Riveiro, M. J. (2011), Visual analytics for maritime anomaly detection, PhD thesis, Örebro University.

Statheros, T., Howells, G. & McDonald-Maier, K. (2008), 'Autonomous ship collision avoidance navigation concepts, technologies and techniques', *The Journal of Navigation* **61**, 129–142.