AIS Data as Trajectories and Heat Maps
Andreas S. Andersen, Andreas D. Christensen, Philip Michaelsen, Shpend Gjela, Kristian Torp
Aalborg University
Denmark
{asan18,adch18,pmicha18,sgjela18,torp}@cs.aau.dk

ABSTRACT
All large ships are by international law required to provide their position, speed, and course while sailing. This data is called AIS data. Several maritime organizations make this data freely available. In this paper, we present two approaches to querying AIS data. The first approach combines the individual AIS data records into trajectories and the second approach is to combine many trajectories into heat maps. The first approach is well suited, e.g., to find the complete route of a few ships or study how many ships are navigating in a smaller area known to be complicated to sail. The heat-map approach is particularly well suited to provide an overview of ship movements in large areas. For the trajectory approach, we introduce and define a novel way to query AIS data called a trident query. This query type is developed in close collaboration with domain experts. The core idea with a trident query is to visualize route choices. The heat-map approach works both for user-defined areas and for predefined Areas Of Interest (AOI) cells. The trajectory approach is difficult to scale and we show how the trajectories can be simplified to make querying and visualization more efficient. We present data on a map and statistical details are provided in graphs and tables, e.g., the distribution of ship types and ship dimensions (length, width, and draught). End-users can filter on attributes such as ship IDs, ship types, and ship dimensions for both the trajectory and heat-map approaches.

CCS CONCEPTS
• Information systems → Information integration.

KEYWORDS
AIS, trajectory, path query, heatmap

ACM Reference Format:

1 INTRODUCTION
The free availability of very large datasets from the Automatic Identification System (AIS) has created a huge interest in analyzing such data to look for patterns in how ships move [1, 5, 7, 12]. A large part of international goods are transported by ship, so both companies and countries are generally interested in more detailed knowledge about sailing routes. For the sailors, the AIS data increases safety at sea because additional information is available, e.g., in search-and-rescue missions.

In this paper, we look at AIS data both as individual trajectories (trips) and as density heat maps. These two approaches are illustrated in the two screen dumps in Figure 1. Figure 1a shows the trajectories that intersect a user-defined polygon. Figure 1b shows the same trajectories illustrated as a density heat map.

The trajectory approach is good if a few ships are followed in detail or if sailing patterns for smaller areas are examined. The heat map approach has the advantage that it can show patterns in very large AIS datasets.

AIS is widely used. The authors of [12] process AIS data in two predefined areas for harbors in the Netherlands and China. A comprehensive guide to navigating maritime informatics such as AIS data is presented in [1]. The book covers many aspects of preprocessing, storing, analyzing, and visualizing AIS data. Cleaning AIS data is covered in [5, 7] and visualization in [3, 11]. Handling moving data is covered in [13]. [6, 9] focus on using AIS data for surveillance to detect patterns (and anomalies). The connection between AIS data and air pollution is covered in [2].

The system presented in this paper, allows the user to query AIS data using arbitrary polygons and present the result as trajectories or heat maps. In particular, we define and implement a new query type called trident that is developed in collaboration with domain experts. The trident query allows users to study sailing patterns in larger areas using trajectories.

The remaining part of the paper is structured as follows. In Section 2 the data foundation and the query types are introduced. Section 3 provides implementation details. Section 4 presents the
2 QUERYING AIS DATA

In this section, we first introduce the AIS data foundation. Next various required definitions are presented, followed by how to present the query result as trajectories or heat maps.

2.1 Data Foundation

AIS is a standard for maritime identification. It allows for communication between ships and can assist in navigation and collision avoidance [10]. The AIS data used in this paper is available online in a CSV format and has 26 columns. This includes columns such as Type of mobile, MMSI (ID of the ship), Latitude, Longitude, SOG (Speed Over Ground), and Ship type. An AIS transponder sends data every 2 to 10 seconds when the ship is underway [3]. We use AIS data from Denmark in total 71101 trajectories from 85M AIS records. The implementation supports AIS data from all over the world.

2.2 The trident Query Concept

The domain experts at The Danish Geodata Agency (DGA) expressed the wish for a new query type. The concept of this query can be seen in Figure 2. This figure shows six trajectories that intersect the dotted line at Level 1. From there, they must intersect at least one dotted line at Level 2. The result consists of the five (blue) trajectories. This concept allows the user to visualize where ships sail that cross a number of user-defined reference lines.

In general, the query is for trajectories that cross a primary line and later cross one or more secondary lines. The order of crossing is irrelevant and as such, the trajectories may cross either level before the other. Due to the maritime domain, the query concept is called a trident query.

![Figure 2: The trident Concept](image)

2.3 Trajectory Formalized

Before defining the trident query we define a point and a trajectory. A point $p$ is a tuple consisting of MMSI, latitude, longitude, timestamp, and SOG. Therefore $p = (MMSI, latitude, longitude, timestamp, SOG)$ and $p \in P$, where $P$ is the set of all points. The dataset used contains 26 columns, in the definitions, we focus solely on the required columns.

A trajectory $t^j_k$ is a collection of points that are sorted by the timestamp, $x$ is a unique ship identifier, and $j$ is the $j$-th trajectory for ship $x$. $t^j_k$ is defined below where all points $p_i$ have the same MMSI and where the time difference between any two consecutive points $p_i$ and $p_{i+1}$ is below 15 minutes. To split the points (AIS records) from a single ship into multiple trajectories when a ship makes a stop, the SOG must be larger than 0.2 knots/hour.

$$ t^j_k = \{p_1, ..., p_n | p_\ell \in P \land 0 < i \leq n $$

$$ \land p_\ell . timestamp <= p_{\ell+1}.timestamp $$

$$ \land TimeDiff(p_\ell, p_{\ell+1}) < 15 $$

$$ \land p_\ell . MMSI = x \land p_\ell . SOG > 0.2 \} \quad (1) $$

2.4 The trident Query Definition

The trident query is executed on a set of trajectories $T$ and uses a set of user-defined lines ($UDL$). This set contains the primary line, $L_1$. The rest of the user-defined lines $L_2, ..., L_k$ are secondary lines, i.e., $UDL = \{L_1 \cup ... \cup L_k\}$. The function $\text{intersects} (\text{geom}_A, \text{geom}_B)$ is defined as $\text{ST_intersects}$ in PostGIS [8]. Finally, the trident is defined in Equation 2 as the set of trajectories from $T$ that fulfill the condition Logic. Equation 3 defines the logic used to filter trajectories, i.e., a trajectory $t^j_i$ must cross the primary line $L_1$ and one-or-more of the secondary lines $L_2 \ldots L_k$.

$$ \text{trident}(T, UDL) = \{ t^j_i \in T \land \text{Logic}(t^j_i, UDL) \} \quad (2) $$

$$ \text{Logic}(t^j_i, UDL) = \begin{cases} 
\text{true} & \text{if } \text{intersects}(t^j_i, L_1) = \text{true} \\
\land (\text{intersects}(t^j_i, L_2) = \text{true} \\
\lor \ldots \lor \text{intersects}(t^j_i, L_k) = \text{true}) \\
\text{false} & \text{Otherwise} 
\end{cases} \quad (3) $$

2.5 Heat Maps

Heat maps are provided in two different ways. By using predefined sea areas called Area Of Interest (AOI) cells or by user-defined polygons. Both ways are handled identically. The following description therefore only focuses on the AOI-cell way.

To create a heat map, an AOI cell is split into smaller areas, e.g., to identify in which parts of the AOI cell the ship density is the highest. The PostGIS method $\text{ST_SquareGrid}$ [8] is used to create a uniform grid that splits an AOI cell. When $\text{ST_SquareGrid}$ is called, a set of grid cells is returned. All of these grid cells make up a bounding box that covers the entire AOI cell. This set of grid cells is used to find all trajectories that intersect with smaller areas of an AOI Cell.

This grid concept is visualized in Figure 3. The black box represents an AOI cell and the dashed grid shows the set of grid cells. The orange arrows represent trajectories that intersect a number of the grid cells. The density in the heat map is based on the number of unique trajectories in each grid cell.

3 IMPLEMENTATION

The PostgreSQL DBMS with the extension PostGIS is used for storing the AIS data. The front-end is built in JavaScript and all server logic in Python using the Django framework [4]. In the following, the database design is described. This is followed by how line simplification is used to speed up the visualization of trajectories.
AIS Data as Trajectories and Heat Maps

3.1 Database Design

The database consists of three tables named traj, ship_data, and traj_detail as shown in Figure 4. The complete trajectory is stored as a linestring in the column geom in the traj table. Other static values for a trajectory such as the length and width of the ship are stored in the ship_data table. The third table, traj_detail, keeps track of variable information for each trajectory such as SOG and draught.

To speed up queries that rely on entries in a date range a B-tree index is created on the column date_of_trip.

3.2 Trajectory (Linestring) Simplification

An examination of the AIS data shows that ships stay on the same course for considerable periods. The time needed to render trajectories (linestrings) to show them in web applications depends on the number of points in the linestrings.

To improve the performance and usability of the interactive map, PostGIS has the function ST_Simplify [8] that simplifies geometries by reducing the number of points in the geometry. The function has a tolerance parameter, which specifies how much the geometry can be simplified. To determine how the accuracy is affected, and whether the simplified trajectories are structurally similar to the non-simplified trajectories, the Fréchet distance is calculated which determines the similarity between the two linestrings (simplified/non-simplified) [8].

Table 1 shows the different results of using ST_Simplify on 4015 trajectories when varying the tolerance value. This value is specified in degrees. Based on the column Fréchet increase in Table 1 the benefit of ST_Simplify flattens at a tolerance value of 0.01°. Here approximately 90% of points are removed from trajectories.

Figure 5a shows a set of trajectories in full detail and simplified (tolerance 0.01°). The two figures are almost identical when visually inspecting them also when zoomed into a smaller area.

4 USE CASES

The system described in this paper is fully implemented and available online at www.ais.boxn.dk. In this section, we describe three typical use cases. These can all be tested online using the URL provided.

4.1 Use Case: Querying Trajectory Data

In the first use case, the primary line L1 is drawn from the coast of Skagen, Denmark, and into the Skagerrak sea. The secondary line L2 is drawn at the Great Belt Bridge while the secondary line L3 is drawn at the Øresund Bridge. These locations are all major maritime traffic hot spots. The result of the query is shown in Figure 6. The query does not default consider the direction of movement, which means it includes (a) the trajectories that cross line L1 first and later either L2 or L3 and (b) the trajectories that cross either L2 or L3 first and later the line L1.

Each trajectory displayed is given a color based on their ship type. Blue is Cargo, yellow is Tanker, orange is Passenger, and grey is Undefined. The arrows indicate the direction of movement.

A panel on the dashboard shows various trajectory statistics for each line L1 to L3 (not shown in Figure 6, found under the tab Statistic at www.ais.boxn.dk). This information includes how many ships passed each line and from which direction.

Directions are fully supported in the implementation and will be demonstrated. The green and red circles on the enlarged area in Figure 6 shows how direction can be queried. In the user interface it can be specified if the direction of the trajectories displayed should be red → green, green → red, or both.

4.2 Use Case: Heat Maps

The next use case shows the density of trajectories around the two major harbors at Hirtshals and Frederikshavn, as seen in Figure 7. The density of ships is higher inside the harbors. Again the polygon used for the heat map can be user-defined or a predefined AOI cell. Several filters are available for all use cases, e.g., filter by

Table 1: Test of Tolerance Values using Fréchet

<table>
<thead>
<tr>
<th>Tolerance °</th>
<th>% Remaining</th>
<th>Avg. Fréchet result</th>
<th>Fréchet increase</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.00001</td>
<td>41.786</td>
<td>0.0025</td>
<td>0%</td>
</tr>
<tr>
<td>0.0001</td>
<td>18.535</td>
<td>0.0127</td>
<td>400.130%</td>
</tr>
<tr>
<td>0.001</td>
<td>12.715</td>
<td>0.0350</td>
<td>176.713%</td>
</tr>
<tr>
<td>0.01</td>
<td>11.437</td>
<td>0.0737</td>
<td>110.253%</td>
</tr>
<tr>
<td>0.05</td>
<td>11.185</td>
<td>0.1021</td>
<td>38.521%</td>
</tr>
<tr>
<td>0.1</td>
<td>11.154</td>
<td>0.1156</td>
<td>13.263%</td>
</tr>
</tbody>
</table>
time period, ship type, MMSI, IMO, as well as the length, width, speed, and draught. Using these filters a harbor can be classified as mostly for fishing ships or for cargo ships. The use of filters will be demonstrated including filtering on directions as discussed in subsection 4.1.

4.3 Use Case: Directional Traffic in Hot Spot

The last use case is shown in Figure 8 and visualizes all trajectories for two days in a small area with very dense traffic. The lines, L1 and L2 are drawn off the coast of Skagen in a northern direction. The lines each cover two widely used ship routes in the area. The majority of ships going East should cross the line L1. Similarly, the ships that go West should cross the line L2. The query confirms that ships follow this sailing rule. In this use case, both L1 and L2 are considered primary lines. In the web solution, this is implemented as cross at least one instead of the default cross first and a second.

The query shows all the trajectories where the ship type is either Tanker, Cargo, and Passenger. We will demonstrate that, e.g., the Fishing ship type follows different routes in this area.

5 SUMMARY

In this paper, we have presented how AIS data can be visualized both as trajectories and as heat maps and that both approaches are highly useful. The system presented is fully functional and available online at www.ais.boxn.dk.

We introduced the trajectory-based trident query that allows end-users to examine the route choices in larger areas. This query type was developed in collaboration with domain experts.

Future work includes describing how AIS data is cleansed, looking at the pros/cons of alternative trajectory definitions (split at a stop or not), and scaling the solution to years of AIS data.

ACKNOWLEDGEMENT

Thanks to the Danish Geodata Agency for data and technical discussion on how to define the trident query.

REFERENCES