Trajectory Modeling for Vessel Tracking and Collision Avoidance

XING YIFAN National University of Singapore A0105591J

Supervisor Professor Ng Szu Hui, Dr. Giulia Pedrielli

Abstract: This project targets establishing a trajectory model for vessel tracking and collision detection and avoidance for the Singapore Port. As large amounts of raw tracking data of vessels such as latitude, longitude, speed, course over route etc., have been captured over the years from AIS system in Singapore port, it would be interesting and potentially significant to study the vessel trajectory from a data science perspective. In particular, the large amount of historical data provide us with the needed data grounding to study vessel motion patterns using machine-learning techniques. In this project, vessel trajectory modeling is essentially divided into two steps. Firstly, a hierarchical-clustering is expected to apply to the vessel trajectories of the same vessel type to extract the representative vessel trajectory of that vessel type. These representative vessel trajectories will describe the behavior pattern of the vessels of a certain type. Secondly, a vessel interaction model would be added based on the behavioral pattern information from the first step, where a probability model will be used to determine which behavior (representative trajectory) that a certain vessel will take on. Finally, it is expected that the developed trajectory model would be used as input to a collision detection and avoidance simulator to study the efficiency and effectiveness of this approach of model vessel trajectory from a data science learning perspective.

Keywords: Vessel Trajectory Modeling, Clustering, Prediction, Interpolation, and Probability Model.

1. Introduction

As the vessel traffic at Singapore Trait is increasing drastically, the number of collision incidences and accidents are increasing significantly as well. Singapore Trait is estimated to have around 1000 vessels at any time at the port [1]. Though the average collision frequency is kept at low of 1.75 per year, the number is increasing [2]. In addition, even though with the assistance of the AIS

(Automatic Identification System) system for vessel data collection and analysis, still, much could be improved on the navigation control process. Specifically, large amounts of historical information of vessel maneuvering are available from the AIS system over the past two years. This large amount of historical data could help us identify behaviors of the vessels. Furthermore, with the assistance of machine learning algorithms, different vessel behavioral patterns could be extracted for different vessel types. In particular, many features could be utilized in the learning procedure, for instance, the categorical static data fields of the vessels such as type, vessel size, engine information etc. and the dynamic data fields such as speed, latitude, longitude, course over route, true heading etc. The behavioral patterns could then be further compensated with an interactional model among the vessels in a neighborhood. Such an interaction model is essential because when vessels come together into a congested region, it is unlikely that each of them will remain on their original behavioral pattern motion. With the behavioral pattern and interaction pattern of the vessel trajectories established, these trajectory models can then be used further as an input to a path planning model to adjust trajectory for both the current target vessel and its neighboring vessels for collision avoidance.

2. Objectives

(1). Development of a data science learning based model for vessel trajectory

- Extraction of vessel behavioral patterns
- Incorporation of vessel interaction model and probability model for pattern selection

(2) Integration of the trajectory model to the path-planning model and further feed these models as input to a simulator to compare the efficiency and effectiveness of this proposed approach in the paper with some existing methodologies like Partially Observed Markov Decision Process.

3. Literature Review

In this section, the definition of a vessel trajectory in the current literature pool is discussed and several existing methodologies in trajectory modeling and motion learning/prediction are examined.

3.1 Vessel Trajectory Definition

In the majority of the current literacies in vehicle motion learning [7] [8] [9] [10], a vehicle trajectory is defined a set of discrete data points at different timestamps, where each of these timestamp points is a vector of the features describing the vessel at that timestamp, such as latitude, longitude, speed etc. The following shows the standard definition of a trajectory Qi for vessel i over a time period T.

$$Q = (d1, d2, d3, \ldots dt, \ldots), dt \in [0, T]$$

$$dt = \begin{pmatrix} x \\ y \\ v \\ \alpha \\ \ldots \end{pmatrix}$$

3.2 Vehicle Motion Learning Algorithms

While there are many motion learning techniques available in the state of art for trajectory modeling and prediction, most of them lie into two broad categories. The first one is about learning the patterns of how vehicles move over time. The second one is dynamic based and usually, it predicts the vehicle motion by typically estimating the current state of the vehicle and then propagating future states with the assumption of a certain fixed model of operation.

3.2.1 Pattern learning Based

There are further two general approaches for pattern based learning method. The first is clustering based and the second is discrete-space based approach. In the first category, representative trajectories of clusters of similar trajectories are extracted and used for motion prediction. In the second approach of discrete-space based approach, usually a Markov Decision Process (MDP) is involved where the current object/vehicle state evolves to the next state with a learned transitional probability distribution.

3.2.1.1 Clustering Algorithm based

a. Trajectory Dissimilarity Measure

As many of the clustering based trajectory learning algorithms rely on algorithms that compare the trajectories. Thus, it is very important to have an effective approach of defining the dissimilarity between trajectories. In [8], the dissimilarity matrix used is simply the average Euclidean-distance (l2 distance) between the corresponding discrete data points along the trajectories. Though it gives experimentally good results for the scenario of a simulated pedestrian environment in [8], more robust trajectory dissimilarity measures are proposed by [4]. In [4], two more novel trajectory dissimilarity measures are proposed. The first is a geographic distance w.r.t center of mass of the trajectory, displacement of end points of trajectory and length of trajectory, which is defined as following (**lltrall** refers to length and **s** refers to the displacement vector of the trajectories).

$$
\begin{aligned}
&geoDist(tra_1, tra_2) = ctrDist(tra_1, tra_2) \\
&+ ctrDist(tra_1, tra_2) \times \frac{|\ \|tra_1\| - \|tra_2\|\ |}{\max(\|tra_1, \|tra_2\|)} \\
&- \operatorname{avg}(\|s_1\|, \|s_2\|) \times \cos(s_1, s_2)
\end{aligned}
$$

The second measure proposed is the semantic distance between the two trajectories, captured by the length of the longest common subsequence of the two trajectories.

b. Trajectory Clustering

In [8], motion learning is achieved by first extracting out the trajectory prototypes that are representative trajectories of the different trajectory clusters. An agglomerative clustering is involved in the process and the mean of one cluster constitutes the representative trajectory for that cluster. The dissimilarity measure used is average Euclidean distance as follows, where di and dj are two trajectories of discrete timestamp based data points. Further, mean and standard deviation of all trajectories within one cluster are extracted as trajectory protocol.

$$
\delta(d_i, d_j) = \left(\frac{1}{\max(T_i, T_j)} \int_{t=0}^{\max(T_i, T_j)} (d_i(t) - d_j(t))^2 \, dt \right)^{1/2}
$$

$$
\mu_k(t) = \frac{1}{N_k} \sum_{i=1}^{N_k} d_i(t) \quad \sigma_k = \left(\frac{1}{N_k} \sum_{i=1}^{N_k} \delta(d_i, \mu_k)^2 \right)^{1/2}
$$

Afterwards, for trajectory prediction, a partial distance of the current vehicle along a trajectory prototype is calculated and a Gaussian model is used to determine the likelihood that the current vehicle will follow that specific trajectory prototype's cluster. Thus, this likelihood determines the trajectory prediction. The partial distance and the likelihood calculation are as shown follows, where dp is the partial trajectory and Ck is the cluster of trajectories represented by ($\overrightarrow{\mu_k}$ and σ_k).

$$\delta_p(d_p, d_i) = \left(\frac{1}{T_p} \int_{t=0}^{T_p} (d_p(t) - d_i(t))^2 \, dt \right)^{1/2} \qquad P(d_p \mid C_k) = \frac{1}{\sqrt{2\pi\sigma_k}} e^{-\frac{1}{2\sigma^2}\delta_p(d_p,\mu_k)^2}$$

In [9], the clustering process is enhanced and made more robust by breaking down the trajectories into sub-trajectories. Further, a Hidden Markov Model is fitted in [9] for the motion prediction and planning.

c. Expectation Maximization

Another similar approach that extracts protocol trajectories works based on an Expectation Maximization algorithm instead of an agglomerative clustering algorithm. The Expectation that EM algorithm works to optimize is the expectation of an indictor variable of whether the current trajectory corresponds to a certain motion model. This is essentially maximizing the likelihood of a trajectory following the motion model by adjusting the motion model (θ_m) step by step. Below is the likelihood that trajectory di following the motion model θ_m up to time t.

$$p(d_i \mid \theta_m) \quad = \quad \prod_{t=1}^{T} p(x_i^t \mid \theta_m^t)$$

More specifically, with the indicator variable c_{im}, which will be 1 if and only if trajectory di is corresponding to the motion pattern θ_m, the joint likelihood becomes the following (with underlying Gaussian probability model).

$$p(d_i, c_i \mid \theta) \quad = \quad \prod_{t=1}^{T} \prod_{m=1}^{M} \frac{1}{\sqrt{2\pi\sigma}} e^{-\frac{1}{2\sigma^2} c_{im} \|x_i^t - \mu_m^t\|^2}.$$

EM algorithm works on the expectation of the total likelihood over all trajectories, which is given by the product of the individual joint probabilities as shown above. Thus, when EM converges, we will have a set of θ_m's in the form of μ_m's as the set of protocol trajectories that the vehicle will take. One drawback of this approach is the requirement of the definition of a set of priors for the motion model ($\theta_1^0, \theta_2^0, \dots, \theta_M^0$) and such selection of the priors might affect computational time drastically.

d. Neural Networks

In [11], instead of looking for sequences of discrete timestamp points as protocol trajectories, protocol vectors/points are instead extracted based on a network vector quantization algorithm. The protocol vectors are of the form (x, y, dx, dy), which captures the geographic coordinate and the speed along the two coordinates while input vectors along a trajectory are **xi**'s. Then, the following output response is defined and **mi(t)** is the protocol vector that is closest to **xi(t).**

$$O_i(t) = 1 - \frac{||\mathbf{x}(t) - \mathbf{m}_i(t)||}{\sqrt{N}}$$

Further, with this response defined. A leaky neuron that retains certain past information is used to determine the probability density distribution of the flow of the generated protocol vectors. The leaky neuron is defined as following where **I** is the optimal response of the input with respect to all protocol vectors at time t and γ a coefficient between [0,1] that determines how leaky the neuron is, i.e., governs the rate of decay and thus the memory span of the neuron.

$$a(t+1) = \begin{cases} I & \text{if } I > \gamma a(t) \\ \gamma a(t) & \text{otherwise} \end{cases}$$

Thus, the activation sequence of the leaky neuron defines the sequence of flow vectors, which is the expected trajectory that the vehicle will take.

3.2.1.2 Discrete-space based

Markov Decision Process In a discrete-space based pattern-learning approach for motion learning and prediction, a Markov Decision Process is usually involved. In [6], Foka uses a POMDP (Partially Observable Markov Decision Process), where the unobservable state space S corresponds to a discrete square area of the environment's occupancy grid map around the current vehicle. Then, this POMDP is used to learn the state transitional probability distribution. Further, in Foka[6]'s work, the POMDP is further decomposed to POMDPS with smaller state space. This hierarchical reduction significantly improves the computational time needed. However, one drawback to such approaches are the large state space in general which will often result in a unrealistic computational time for real world applications.

3.2.2 Dynamic based

Kalman Filter Often, dynamic base motion learning approach will assume a fixed model of operation in propagating future states/trajectory points such as Kalman Filter. In [5], the authors made use of an extended Kalman Filter that is compatible for non-linear system for the trajectory feature-state prediction. Furthermore, in [12], an Interacting Multiple Model Kalman Filter (IMM-KF) is introduced, where observations update a bank of Kalman Filters and the one closest to the current trajectory feature values will be used. However, one drawback of such Kalman Filter based approaches is that they only consider the dynamic properties of the vehicle and thus are only suitable for short term prediction and performs poorly for long term prediction due to the failure to incorporate external factors like obstacles and pattern of change in the input trajectory points.

4. Algorithm Design

As it is desirable to have pattern predictions in the long term and learning from the large historical data, I adopted the pattern learning approach, in particular the clustering approach in extracting the trajectory motion patterns of the vessels. Further, as vessels usually stop at certain geographic locations for a long time, for instance, port entries, I further break down the overall trajectory of vessels into trajectories between end points. A hierarchical clustering is applied to extract the protocol trajectories between end points. Afterwards, for probability estimation of the patterns and protocol trajectories, firstly, a simple method of looking at the frequency of the trajectories within the cluster that protocol trajectory represent could be adopted. Furthermore, I could also use the method in [8] to examine the partial distance and the likelihood that a vessel would follow a certain protocol trajectory. Lastly, as discussed with Dr Giulia Pedrielli, the extracted pattern model with its probability distribution could be combined with Boids algorithm to simulate the interaction behavior between the vessels. The following chart summarizes the flow of the algorithm.

Data cleaning for AIS system

↓

Trajectory Pattern Learning and Extraction -> Protocol Trajectories

- Extraction of End points where vessels stay for a consibdearble long time period.
- Extraction and interpolation of the trajectory points between end points
- Agglomerative clustering of the augmented/interpolated trajectories between end points

↓

Probablity density estimation of Protocol trajectories

- Trivial method of looking at frequecy of trajectories in the cluster that protocol trajectory represents
- Gaussian model with partial observed dissmilarity w.r.t the different protocol trajectories.

↓

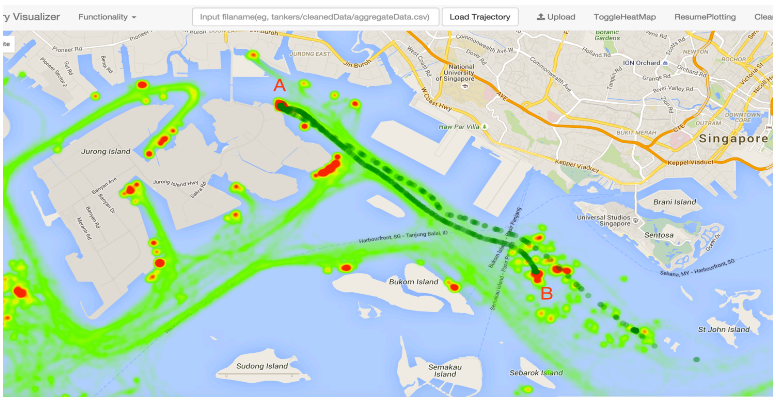
Verification of effectiveness with simulation

- Compensate the trajectory pattern model with interaction model when the vessels are close to each other
- Feed the model as the input to the path planning model

5. Current Progress and Result Analysis

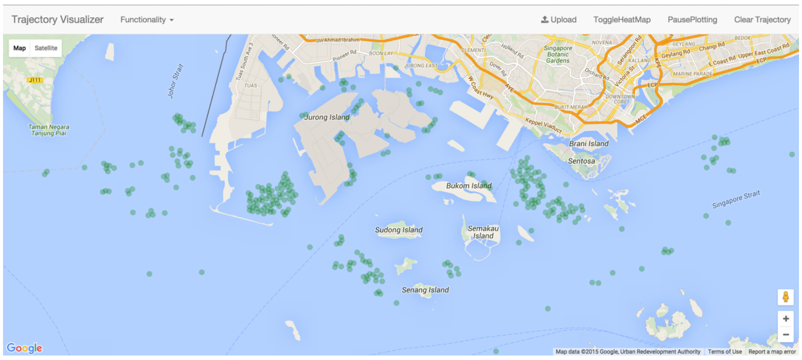
5.0 Data cleaning and Vessel trajectory visualization

I have completed the majority of the cleaning work and I also built a web-interface based map view for better visualization of the vessel trajectories. An example is shown as the following, where A and B are two end points.



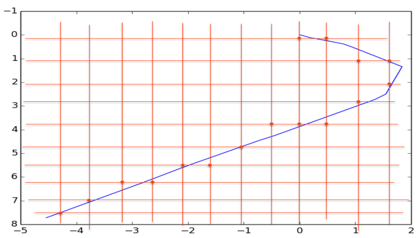
5.1 End point extraction

For end point extraction, I aggregated data for one vessel type for the testing purpose. I will categorize a geo-location with a circular neighborhood of 200 meters as an endpoint if a vessel stays in that region for more than one hour. An example is shown as follows, where the green circular regions are the end points for an aggregated data of 10 tankers.



5.2 Vessel trajectory interpolation for clustering purpose

After the endpoints are extracted, trajectories need to be augmented so that they could be standardized for clustering purpose. For the augmentation process, I have tried out spline and cubic interpolation. Furthermore, there can be two ways of interpolation. The first being interpolation based on purely geographic information, i.e., interpolate to get the equal distance (x,y) coordinates of the trajectories. This can be non-trivial for the case where trajectory is of circular shape and cannot be nicely represented as a simple 1D function of x coordinates. In this case, I propose using grid approximation in taking the sample points along the trajectory and I am in the process of experimenting around the robustness of this approach. (elaborated as below).



Second way of interpolation is relatively more straightforward, where equal temporal/time interval points are interpolated along the trajectory.

5.3 Vessel trajectory Hierarchical Clustering

As I want to avoid defining some priors for the clustering purpose, the current way of cluster the trajectories is via a hierarchical clustering since in Hierarchical clustering, there is no need for a pre-defined number of clusters like in K-means clustering, instead, just a termination condition is needed.

6. Future Work

6.1 Vessel Interactions and Prediction Probability Model

With the clustering and extraction of the protocol trajectories done, the probability distribution of these trajectories could be firstly, in a naïve way, be estimated by the frequency of the trajectories within that certain cluster the protocol trajectory represents. Secondly, if we have some information on the vessel's movement from the starting point, the distance and pattern that vessel travels from the starting point to the current point can be used to compute the partial distance and the likelihood that a vessel will take on a certain protocol trajectory pattern as shown in [8].

Furthermore, as discussed with Dr Giulia Pedrielli, a Boids algorithm can be applied to simulate the vessels' interaction behavior. The idea is that when the vessels are far apart, they will follow their most likely protocol trajectory pattern and when they are close, the group of vessels in a neighbor will be treated as a flock of 'birds' and follow the movement pattern simulated via Boids algorithm.

6.2 Integration with Collision detection and avoidance simulator

Lastly, integration with the collision detection and avoidance model is expected to examine the effectiveness of the approach of learning vessel motion from a data science perspective from large historical data. It is expected that the combined model would be fed into a simulator to compare and contrast with some existing methodologies based on Markov Decision Process.

7. References

[1] Facts and Trivia. (2009.). Retrieved December 11, 2015, from http://www.mpa.gov.sg/sites/maritime_singapore/what_is_maritime_singapore/other_facts_you_may_not_know.page

[2] Weng, J., Meng, Q., & Qu, X. (2012). Vessel Collision Frequency Estimation in the Singapore Strait. *Journal of Navigation J. Navigation,*207-221.

[3] Carvalhosa, S. (2010). Cooperative motion control of multiple autonomous marine vehicles: Collision avoidance in dynamic environments. *7th IFAC Symposium on Intelligent Autonomous Vehicles (2010).*

[4] Liu, H., & Schneider, M. (2012.). Similarity measurement of moving object trajectories. *Proceedings of the Third ACM SIGSPATIAL International Workshop on GeoStreaming - IWGS '12.*

[5] P. Perera, L, Soares. C, G (2010). Ocean Vessel Trajectory Estimation and Prediction Based on Extended Kalman Filter. *ADAPTIVE 2010 : The Second International Conference on Adaptive and Self-Adaptive Systems and Applications*

[6] Foka, A., & Trahanias, P. (2002.). Predictive autonomous robot navigation.*IEEE/RSJ International Conference on Intelligent Robots and System.*

[7] Vasquez, D., Laugier, C., & Fraichard, T. (2009.). Incremental Learning of Statistical Motion Patterns with Growing Hidden Markov Models.*Springer Tracts in Advanced Robotics Robotics Research,* 75-86.

[8] Vasquez, D., & Fraichard, T. (2004). Motion prediction for moving objects: A statistical approach. *IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA '04. 2004.*

[9] Sung, C., Feldman, D., & Rus, D. (2012). Trajectory clustering for motion prediction. *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems.*

[10] Bennewitz, M., Burgard, W., & Thrun, S. (2002.). Learning motion patterns of persons for mobile service robots. *Proceedings 2002 IEEE International Conference on Robotics and Automation (Cat. No.02CH37292).*

[11] Johnson, N., & Hogg, D. (1995.). Learning the Distribution of Object Trajectories for Event Recognition. *Procedings of the British Machine Vision Conference 1995.*

[12] Mazor, E., Averbuch, A., Bar-Shalom, Y., & Dayan, J. (2002). Interacting multiple model methods in target tracking: A survey. *IEEE Trans. Aerosp. Electron. Syst. IEEE Transactions on Aerospace and Electronic Systems,* 103-123.