2014 International Conference on Advances in ICT for Emerging Regions (ICTer): 078 - 084

A Novel Approach To Automate Surrounding Ships In A Virtual Maritime Environment

W.D.N.Y Gunathilake[#1], N.D Kodikara[#2], C.I Keppitiyagama[#3], K.D Sandaruwan[#4]

[#]*School of Computing, University of Colombo*
35, Reid Avenue, Colombo 07, Sri Lanka

[1]gnaduni@gmail.com
[2]ndk@ucsc.cmb.ac.lk
[3]cik@ucsc.cmb.ac.lk
[4]dsr@ucsc.cmb.ac.lk

***Abstract*— Full mission marine training scenarios are essential in training marine trainees with ship handling simulators. In a good training simulation not only the static environment but also the dynamic surrounding ships plays a major role in terms of behaviour realism. The surrounding ships also need to navigate according to COLREG (Convention on International Regulation for Preventing Collisions at Sea) navigational rules. This research presents a novel approach to automate the surrounding vessels in a marine simulation environment with a central controller to control the behaviours of each surrounding vessel in the marine environment. The controller uses positioning data of real ships which are derived from a set of historical Automatic Identification System (AIS) data and map the positioning data with the surrounding ships in virtual environment to obtain the navigation. An Ant Colony Optimization (ACO) algorithm is used to avoid the collisions by generating a new path when both trainee's ship and surrounding ships are in the vicinity of each other. The collision handling mechanism uses COLREG rules. The ACO algorithm generates successful collision avoided paths in all head-on, crossing and over-taking encountering situations. The number of turning points is used to measure the smoothness of the path and number of obstacles in the environment and number of nodes in the graph affects the smoothness of the path. With these results it is revealed that AIS data can be used as an assistant to automate surrounding vessels in a virtual maritime environment together with a proper collision avoidance mechanism.**

***Keywords*—AIS, Ant Colony Optimization (ACO), collision avoidance, autonomous navigation, surrounding ships, target ships, ship handling simulators.**

I. Introduction

Maritime Training and virtual marine simulations are closely allied terms with an intimate relationship which has strengthened through rapidly growing computing power and technologies for decades. Naval forces and nautical training institutes all over the world uses maritime simulators to train naval officers over high risky, expensive and dangerous real marine training in real sea environments due to low cost, safety and accuracy.

Due to the large number of collision situations occurs all over the world while navigation, ship handling simulators demand higher usage than any other types of marine simulators in maritime training. It is stated that 75 -96% of the accidents and mishaps in the sea environment occurred due to human errors [1],[2]. And 89 - 96% of those collision situations could have been avoided by proper measures to reduce the collision and by the decisions taken in proper time. Hence, ship handling simulators need higher accuracy and realism with different levels of difficulty and risk.

Recent analysis's [3] on several international and domestic ship handling simulators revealed that most of the simulation systems are suffer from a common shortage of not having the intelligent navigation capability of target/surrounding vessels according to COLREG[4] navigational rules. This shortage leads to the deviations of the understanding and the decision making skills of trainees in actual navigation scenarios. Thus, ship handling simulators requires an accurate blend of virtual reality, artificial intelligence, decision making, autonomous control and the standard navigational rules, in order to achieve better realism of behaviours in terms of navigating surrounding vessels in the virtual environment.

This paper presents our novel approach which is taken to navigate surrounding ships automatically in a virtual maritime environment and its integration with the "Vidusayura" [5] ship handling simulator. In the forthcoming section, the novel approach and its design information are described. Section III and IV describe the implementation details and evaluation results respectively. And section V and VI consists of background work done in this research domain and the research conclusions.

II. System Design

The basic concepts of ship navigation and COLREG navigational rules on basic ship to ship interaction were mapped together to obtain the autonomous surrounding ship navigation. This design introduces two novel approaches which were designed for track generation of surrounding ships and for collision avoidance between surrounding ship and trainee's ship. Three modules were designed as in figure 2 and the design focuses on two main design concerns.

1. Collision Avoidance according to COLREG guidelines

To obtain a better training experience, the trainees should have a better understanding of navigational rules. Hence, the integration of COLREG with the collision avoidance strategy is required.

2. Use historical real human behaviors

Even though COLREG defines the actions to be taken for collision avoiding, the rules do not suggest factors such as when to take actions, what is the ultimate clearance distance between two vessels or do not specify the turning angles. Those measurements left to the human to done by his experience and sense. The most common method that is used

978-1-4799-7732-1/14/$31.00 ©2014 IEEE

to avoid collisions while navigation in simulators is mimicking the real behaviors to some extent. However, they do not represents the real decision making process and the real behaviors. In this research work we are using real behavior of real ships to automate surrounding ships in the virtual environment.

A. *Functional Overview*

According to our analysis done by exploring the background, we identified 5 main steps that should be included in the intelligent autonomous ship handling process as in Figure1..

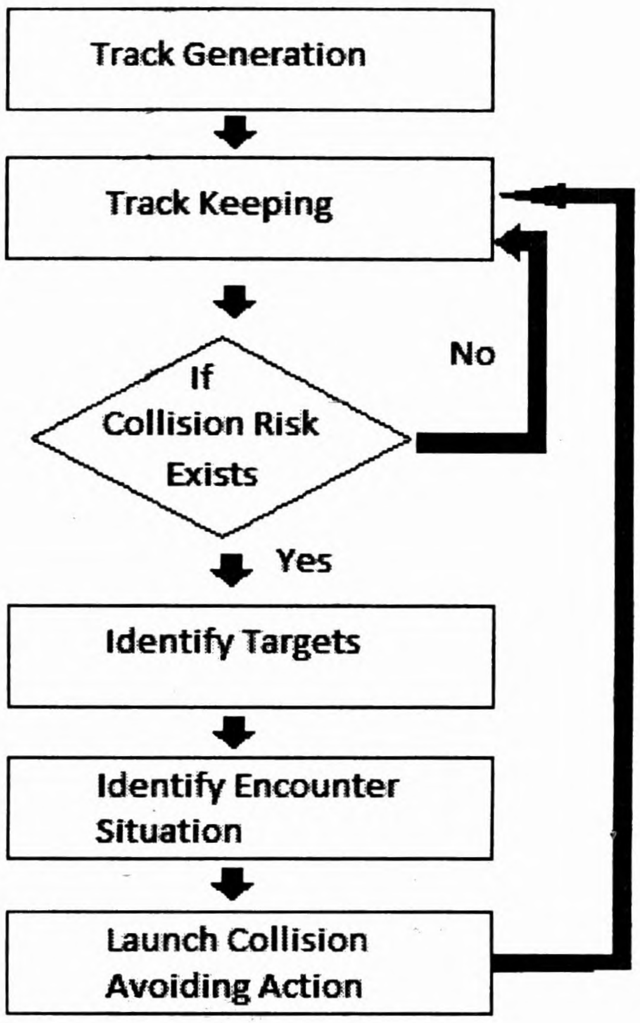


Fig. 1.Abstract Functional Overview

1) *Track Generation*

Track generation is done using an AIS (Automatic Identification System) model. Real AIS data of ships in a real harbor environment were collected. The positional information such as latitude and longitude of each real ship in the real environment were used as the way points to generate tracks of each surrounding ship in the virtual environment. Thus, the autonomous navigation of surrounding vessels was achieved by using a one-to-one mapping of real world within the virtual environment. The collision avoidance between each surrounding vessel is automatically happen within the simulation since the AIS data which collected from the real environment are applied directly to each surrounding vessel. Also AIS information contains the real human behavior of ships. Therefore decisions taken while navigation are real and the navigation complies with the COLREG navigational rules. Thus, without requiring additional computer overhead to perform these actions, AIS model automatically simulates all these functionalities.

2) *Collision Risk Determination*

Even though, the collision avoidance between each surrounding vessel is automatically occurred within the mapping, the surrounding ships which are driven by AIS data do not know that the trainee's virtual ship exists in the system since, the virtual ship does not cooperate with the real world mapping. When the trainee's ship enters to an area of vicinity with the surrounding ship, both ships might have an intersection of their paths which might leads to a collision between surrounding ship and trainee's ship. Hence, the collision risk between trainee's ship with surrounding ships need to be calculated.

An area of observation is defined with the trainee's ship as a circle and the radius "*r*" is given as a user parameter. When the r < DCPA (Distance at Closest Point of Approach) of a surrounding ship with trainee's ship, the surrounding ship is identified as collision candidate with the trainee's ship. If there is a risk of multiple ship collision, the targets and collision risk with each target is calculated and most threatened ship is prioritized.

3) *Encounter Situation Identification*

When two ships are encountered, the encountered situation is identified using the classification proposed by K. Hasegawa in [8]. We have considered head-on, crossing and overtaking situations in this research.

4) *Collision Avoidance Action Identification*

The collision avoidance activity is done by modifying the original AIS track of each surrounding ship using an Ant Colony Optimization (ACO) algorithm which is a swarm intelligence method. The ACO algorithm generates a path which complies with navigational rules and sends the newly generated waypoints. After handling the collision, the surrounding ship will again align with its previous AIS track.

B. *System Architecture*

The following system architecture was proposed as in figure 2 with three modules; *(1). Track Generator (2). Controller and (3). Collision Handler.*

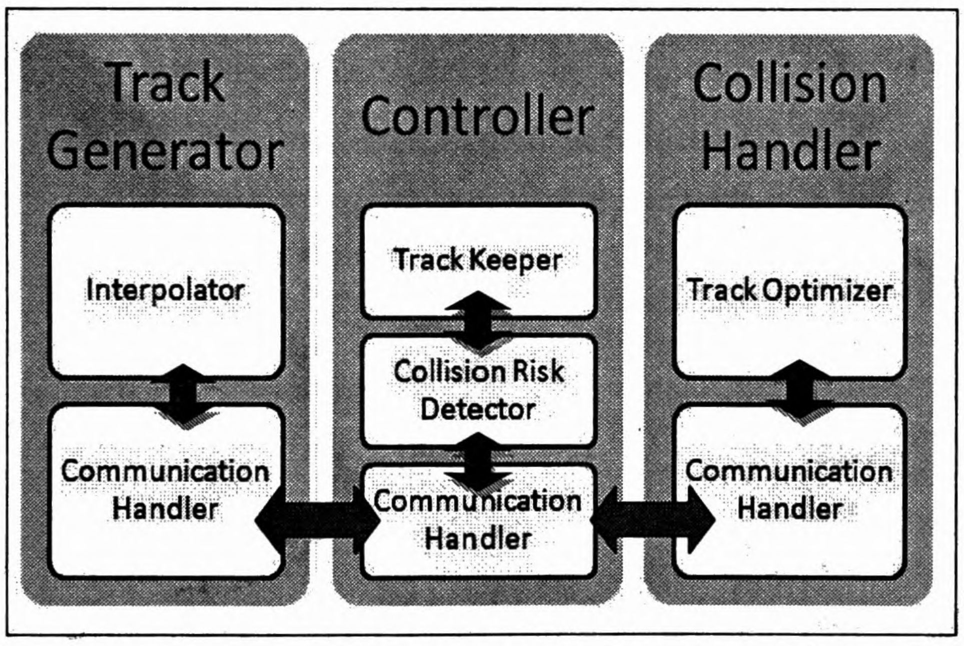


Fig. 2. System architecture of proposed solution

1) *Track Generator*

Track Generator consists of two components; Interpolator and Communication Handler. Interpolator takes latitude, longitude of each ship recorded in the AIS database as the input. AIS way points were used to generate the trajectory of virtual ships in the simulation environment. Since AIS data are transmitted from distance places, the packet losses may happen and the time interval between incoming packets may vary. Therefore, interpolator restores the trajectories of each vessel by using an interpolation technique and produces a smooth trajectory. Communication handler sends the smoothed positional data to the controller component of the system.

2) Controller

Controller component is handling with the server of the "Vidusayura" ship simulator directly. Track keeper is responsible for maintaining the ship on the desired track and it maintains each and every surrounding ship in its track simultaneously. When a ship has a risk of collision the current control of the ship is given to the collision handler component. After creating a collision avoided path, track keeper receives the collision free ship track, keeps the ship on the track and aligns with the previous AIS track. Both collision risk detector and track keeper sub components work collaboratively and simultaneously. Collision risk is calculated when a surrounding vessel enters the circle of the trainee's ship. Then the target ships identity is passed to the track keeper to mutate the track keeping of target ships and also determines the encounter type of ships. The encounter type and the positions are sent to the collision handler component to avoid the collision.

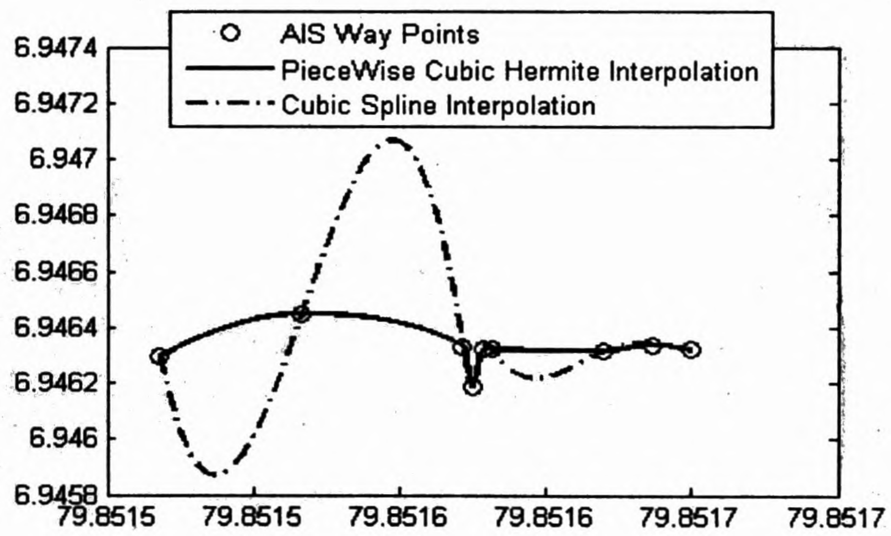
3) Collision Handler

In collision handler component, track optimizer modifies the track of threatened surrounding ship by using the ACO algorithm by avoiding the dynamic and static obstacles in the environment with the corporation of COLREG navigational guidelines. Then, it sends the collision free ship track to the controller for track keeping.

III. SYSTEM IMPLEMENTATION

The system implementation of this research consists of track generating and collision handling between ships by considering single ship-to-ship interaction rules. Several key functionalities of this system will be discussed under this section.

A. Track Generation using AIS Way-points Interpolation



B.

Fig. 3. Results obtained from cubic interpolation to restore AIS track

Track generation is achieved using an AIS model. After analysing AIS positional data, it is observed that the AIS messages have a latency among them which leads to reasonable gaps between adjacent data points. Due to this message loss the direct latitude and corresponding longitude value cannot be applied as the track of each surrounding vessel and it is unable to generate a smooth track for ship to follow. Cubic Interpolation is used to restore AIS tracks of ships within a given time period. We have used "Cubic Spline interpolation" and "Piecewise Cubic Hermite interpolation" techniques and according to the results obtained from both methods as in figure 3, Piecewise Cubic Hermite interpolation tends to provide more realistic and accurate ship tracks.

C. Collision Risk Detection and Encounter Type Identification

The possible collision situations are detected by defining an "area of observation" with the trainee's ship as depicted in the figure 4. The "R" is the radius of the circle which is defined around the trainee's ship as the "area of observation". The value of "R" depends on the trainee's ship type. And instructor can define the value at the beginning of the training session.

The DCPA and TCPA (Time of Closest Point of Approach) with the trainee's ship are calculated with each position update of each surrounding ship. When the R <DCPAs, for the surrounding ship "s", the "s" is selected as a candidate for collision avoidance. When multiple ships are encountered in the area of observation, the ship with the highest collision risk is taken as the most suitable ship to apply collision avoidance action. The ship with the highest collision risk is the ship with the smallest TCPA value. This research work is initiated with the assumption that handling 2-3 ships within the area of observation and considered only single ship collision avoidance situations.

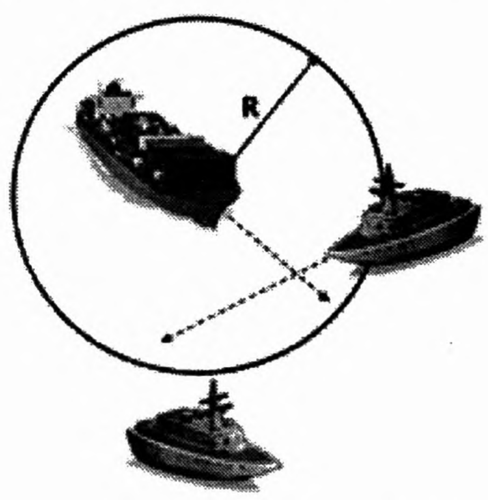


Fig. 4. Collision Risk Detection

The next step is to identify the encounter state of the candidate ships. According to COLREG rules, there are three main encounter types of ships. Encounter type can be identified using two angles between trainee's ship (own ship) and target surrounding ship as described in [8]. This research project adopts the encounter type classification proposed by Hasegawa in[8] as in figure 5. Three main encounter types are considered to avoid collisions and suitable collision avoidance path planning actions are performed according to the COLREG as in table 1 to find a safe and economical path between source and destination.

D. Collision Avoidance Route Planning Using an ACO

Inspiring from the ACO concepts which leads to find a path within a graph, our research approach proposes a ACO algorithm to plan the path within a source and a destination location by avoiding both static and dynamic obstacles and includes the factors of the COLREG rules. ACO can generate optimal paths even within the dynamic environments and also avoids local optimal solution due to pheromone update. It is possible to have more than one path solution between source and destination and ACO produce optimal paths between given source and destination.

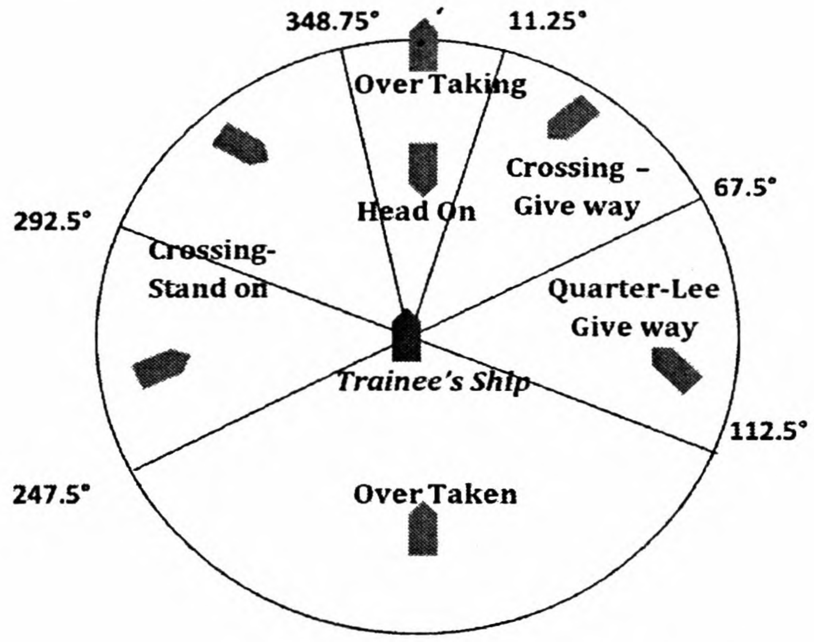


Fig. 5. Encounter Situation Classification

Table 1: Collision avoiding actions of Trainee's ship

Encounter Situation	Avoiding Action
Head on	Right Turn
Crossing Give-way	Right Turn
Crossing Stand-on	Right Turn
Overtaking	Right Turn
Overtaken	Right/Left Turn
Quarter Lee Give-way	Right Turn

1) Representation of the Environment

A typical bi-directional graph is selected to represent the nodes in the environment as a 2D map. There are 100 nodes in the graph, which represents the locations or turning points of the path that is going to generate. Each node has its own x and y axis coordinates, its id, and an indicator to indicate whether it is an obstacle or not. There are n*n nodes which represent the locations as a grid where n can be a positive integer. (For the current solution we assume n > 7, otherwise the number of nodes is not sufficient to include all the environment details in a large sea basin). The collision handler component gets the source location and the destination location which needs to generate the path in between and according to the data, the graph is generated to represent the locations as depicted in figure 6.

2) Traversing the Graph

The ants in an ant colony are most likely to follow the path with more pheromone. Initially, all the edges have same amount of pheromones and after each tour, pheromone levels are updated for each edge according to the length and then ants are again placed in the source and the operation continued. And finally, the shortest and optimal path is selected.

3) Edge Selection

For ant k, the probability p^k_{xy} of moving from state "u" to state "v" depends on the combination of two values; (1). *The attractiveness η_{uv} of the move*, as computed by the heuristic which indicates the priori desirability of that move and (2). *The trail level τ_{uv} of the move*, which indicates how competent it has been in the past to make that particular move.

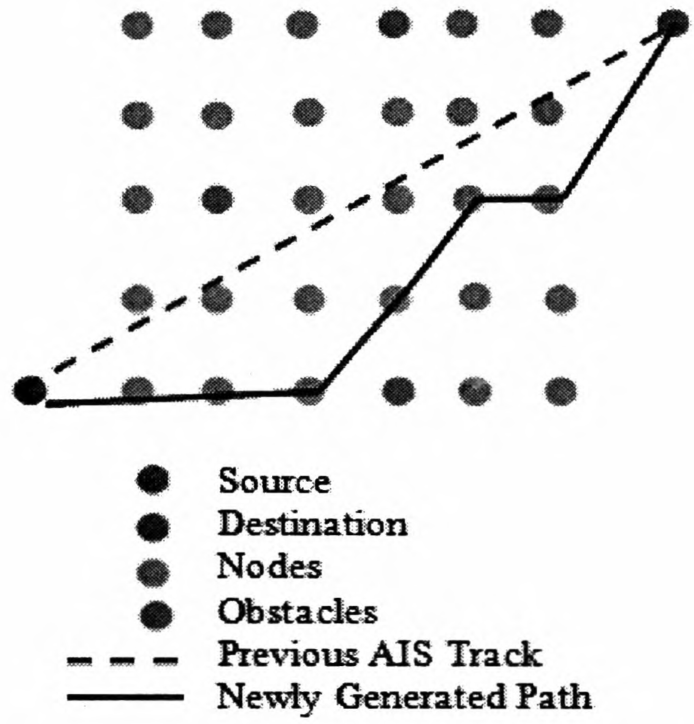


Fig. 6. Graph Representation and Route Generation in ACO

In general, the k[th] ant moves from state u to state v with probability,

$$P_{uv}^{k} = \frac{(\tau^{\alpha}{}_{uv})(\eta^{\beta}{}_{uv})}{\sum_{v \in NotTraversedV}(\tau^{\alpha}{}_{uv})(\eta^{\beta}{}_{uv})} \quad (1)$$

τ_{uv} is the amount of pheromone deposited for transition from state u to v, α is the parameter to control the influence of τ_{uv} or the weight of the pheromone, η_{uv} is the desirability of state transition uv (a priori knowledge, typically $1/d_{uv}$, where "d" is the distance) and $\beta < 1$ is the parameter to control the influence of η_{uv} or the weight for the distance of the edge which can be modeled after visibility. τ_{uv} and η_{uv} represent the attractiveness and trail level for the other possible state transitions.

The edge selection is constrained with several other limitations due to the context that it is using. Since, a collision avoiding mechanism is being developed; both the static and dynamic obstacles are needed to be avoided. The adjacent nodes are selected according to the pheromone levels on each edge, value of the "Is Obstacle" flag and the turning angle of collision avoidance which can be varied in between 30 - 90 degrees as suggested in [33]. For each edge selection of node in i[th] column, the candidate nodes are selected from the (i +1)[th] column of the grid.

4) Pheromone Update

Typically, trails are updated when all the ants have completed their solution. The increment or the decrement of the value of trail corresponds to moves that were part of good or bad solutions, respectively. The pheromone level left on each edge at the end of each tour is calculated using following formula.

$$\Delta\tau_{uv}^{k}(t = \begin{cases} Q/L_k(t), if\ ant\ k\ uses\ curve\ uv\ in\ its\ tour \\ 0, \quad otherwise \end{cases} \quad (2)$$

Here Q is a constant and $L_k(t)$ is the total length of the k[th] ant's tour. Then the above calculated value is used to calculate the total pheromone value after each tour using the following formula.

$$\tau_{uv}(t) = (1-\rho) * \tau_{uv}(t) + \sum_{k} \Delta\tau_{uv}^{k} \quad (3)$$

In this formula, τ_{uv} denotes the amount of pheromone deposited for a state transition uv, ρ denotes the pheromone evaporation coefficient and $\Delta\tau_{uv}^{k}(t)$is the amount of pheromone deposited by k[th] ant. The update happens when all the ants reaches the destination for each iteration.

IV. EVALUATION

The Evaluation is mainly focused on evaluating the path generation of the surrounding ships using interpolation and the collision avoidance activity of the encountered surrounding ships.

A. Data Sampling Plan

Several testing scenarios were created to evaluate results of both novel approaches.

The AIS positional data of 20 dynamic vessels were collected. The vessel tracks were interpolated using cubic spline interpolation technique and piece-wise cubic Hermite interpolation technique. The results were analyzed using behavior of each interpolated ship track.

A code level run time analysis was done in order to identify the running time changes by changing the number of obstacles and the number of nodes in the grid at each time. 20runs were performed for each change in the number of nodes and for each change, the average value of each sample were calculated and taken to the analysis.

The number of turning points that lies within a path mainly affects the sensitivity and the accuracy of the generated path. Hence, the parameters in the algorithm, such as number of nodes, number of obstacles might affect the number of turning points in the generated collision avoided path. Different data sets were collected to analyze the effect of the parameters.

Four or more data samples were collected by changing the one parameter at a time and each sample contains counts of turning points in the newly generated path for 20 different runs of the ACO algorithm.

B. Results Analysis

1) Track Generation with AIS Way-point Interpolation

AIS positional data of 20 different underway ships in the Colombo harbor area are selected. Cubic spline interpolation and piecewise cubic Hermite interpolation is used to interpolate the way points of those data and results were obtained as depicted in the figure 3. The results were analyzed by considering the realistic nature of the path.

In the figure 3, the green continuous line represents the piecewise cubic Hermite interpolation result and the red dashed line represents the cubic spline interpolation. According to results, we observed that when the distance between two way points are sufficiently large, the two interpolation techniques produce different outputs. According to the results of all ships, the piecewise cubic Hermite interpolation tends to produce a more accurate and realistic results than the cubic spline interpolation.

2) Collision Avoided Path Generation

In collision avoidance evaluation phase, the code level running time was calculated by changing number of nodes and number of obstacles. A statistical parametric analysis is performed to identify the parameters that might be affect to the number of turning points within the generated path. The algorithm has executed for 20 different cases of path planning for the following analyzes. The runs were conducted in a Windows7 platform with Core i3 processor.

- *Running Time Vs number of nodes in the graph*

The ACO algorithm is executed 20 times for each change in number of nodes and the averaged results are listed in the following table 2. For all the cases, other parameters were fixed as number of static obstacles = 5, α = 1.0, β = 5.0, and ρ = 0.5.

Table 2: Number of Nodes Vs Running time

Number of Nodes	Average Running time (s) [Heading]	Average Running time (s) [Crossing]	Average Running time (s) [Overtaking]
25	0.02	0.02	0.04
49	1.57	1.64	1.64
81	5.47	5.29	5.39
121	15.19	14.46	15.27

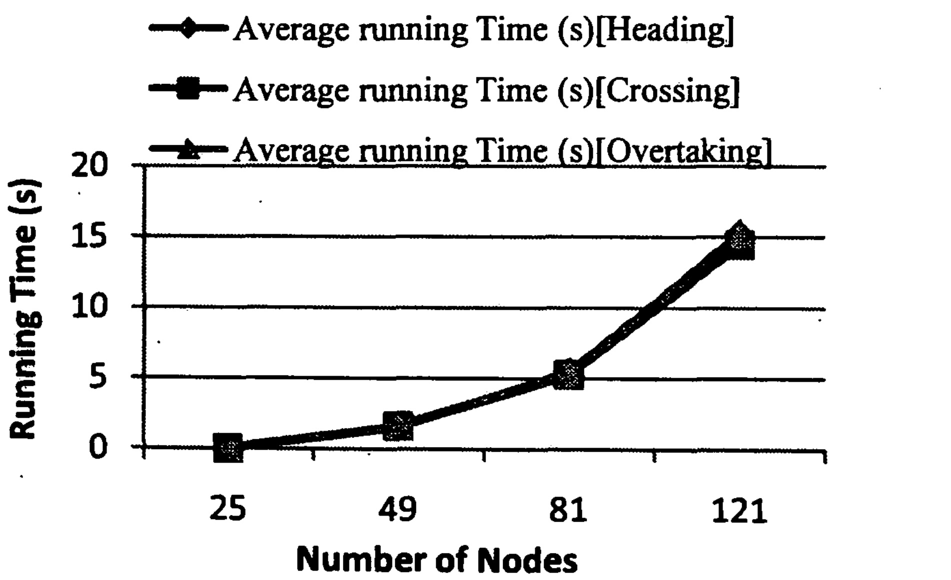


Fig. 7. Running Time Analysis by changing number of nodes in the graph

According to the results in figure 7, for each dataset of three head-on, overtaking and crossing data sets, when the number of nodes increases running time also increased. Also figure 7 depicts that whatever the encounter type of the ship, the time taken to generate a solution is always increasing when number of nodes is increasing.

- *Running Time Vs number of obstacles in the environment*

The ACO algorithm is executed 20 times for each change in number of obstacles and the averaged results are listed in the following table 5.2. For all the cases, other parameters were fixed as number of nodes in the graph = 121, α = 1.0, β = 5.0, and ρ = 0.5.

According to the results of table 3, it is possible to identify that unlike the changing number of nodes, the changes in number of obstacles have slight improvements of running time when the number of obstacles are changing. The algorithm was tested up to 20 static obstacles in the environment. It can be concluded that when number of obstacles are increasing, the running time of the algorithm also increases.

Table 3: Number of Obstacles Vs Running time

No of Static Obstacles	Average Running Time (s)
2	14.87
4	15.27
6	15.82
8	16.04
10	17.89
12	21.89
14	22.13
16	22.77
18	23.01
20	22.90

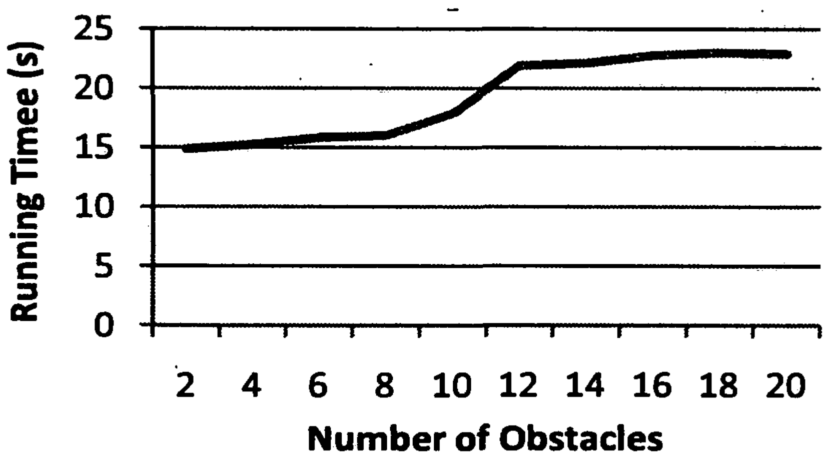


Fig. 8. Running Time Analysis by changing number of obstacles in the environment

Statistical Parametric Analysis

The number of turning points that lies within a path mainly affects the sensitivity and the accuracy of the generated path. Therefore, the parameters in the algorithm, such as number of nodes, number of obstacles, α, β values affect the number of turning points in the generated collision avoided path. Different data sets were collected to analyze the effect of the parameters.

The one-way analysis of variance (ANOVA) is used to determine whether there are any significant differences between the means of collected data samples. The IBM SPSS tool is used to perform the parametric tests. The ANOVA test is performed to identify whether there is a impact on the number of turning points from parameters such as α, β and number of obstacles. The analysis is done using the 0.05 standard significant value.

- *Changing the number of nodes in the graph*

The algorithm is executed by changing the number of nodes in the graph. For each change, the algorithm executed 20 times and the number of turning points was counted. Likewise, 4 different samples with 20 data items were collected where number of nodes equals to 25,49,81 and 121.

The obtained results are depicted in table 4. According to the results, the significant value is 0.000 where; 0.000 <0.05. With this result, we reject the null hypothesis and it is possible to conclude that there is a statistically significant difference between the samples of turning points with different amount of number of nodes.

- *Changing the number of obstacles in the environment*

The algorithm is executed by changing the number of obstacles in the environment. For each change, the algorithm executed 20 times and the number of turning points was counted. Likewise, 7 different samples with 20 data items were collected where number of obstacles equals to 2,3,4,5,6,7 and 10.

Table 4: ANOVA test results for number of nodes parameter

	Sum of Squares	df	Mean Square	F	Sig.
Between Groups	94.600	3	31.533	23.949	.000
Within Groups	47.400	36	1.317		
Total	142.000	39			

Table 5: ANOVA test results for number of obstacles parameter

	Sum of Squares	df	Mean Square	F	Sig.
Between Groups	85.683	6	14.281	2.203	.045
Within Groups	1121.311	173	6.482		
Total	1206.994	179			

The obtained results are depicted in table 5. According to the results, the significant value is 0.045 where; 0.045 <0.05. With this result, we reject the null hypothesis and it is possible to conclude that there is a statistically significant difference between the samples of turning points with different amount of number of obstacles.

- *Changing the α in the ACO*

The algorithm is executed by changing the α value in ACO algorithm. For each change, the algorithm executed 20 times and the number of turning points was counted. Likewise, 14 different samples with 20 data items were collected where α equals to 1,2,3,4,5,6,7,8,9,10,12,15,18 and 20

The obtained results are depicted in table 6. According to the results, the significant value is 0.885 where; 0.885 >0.05. With this result, we do not reject the null hypothesis and it is possible to conclude that there is no statistically significant difference between the samples of turning points within 1-20 α value range.

Table 6: ANOVA test results for α parameter

	Sum of Squares	df	Mean Square	F	Sig.
Between Groups	18.889	13	1.453	.560	.885
Within Groups	690.508	266	2.596		
Total	709.396	279			

- *Changing the β in the ACO*

The algorithm is executed 20 times by changing the β value in ACO algorithm and the number of turning points was counted. Likewise, 6 different samples with 20 data items were collected where β equals to 1,5,10,15,20 and 25.

Table 7: ANOVA test results for β parameter

	Sum of Squares	df	Mean Square	F	Sig.
Between Groups	21.475	5	4.295	1.400	.230
Within Groups	349.850	114	3.069		
Total	371.325	119			

The obtained results are depicted in table 7. According to the results, the significant value is 0.230 where; 0.230 >0.05. With this result, we do not reject the null hypothesis and it is possible to conclude that there is no statistically significant difference between the samples of turning points within 1-25 β value range.

With above results in the statistical analysis, it is possible to conclude that and parameters do not have effect on the number of turning points in the proposed path generation ACO algorithm like number of obstacles and number of nodes in the graph.

V. Background

A behavior-based approach was proposed in 2010 [6], that tried to answer 3 major challenges; (1). Efficiency of the training system (2). Readiness of the response and (3). The realism of behavior. Researchers have developed a Controller Authoring Tool and have modeled 7 primitive behaviors of ships. Each behavior represents its goal and the status of goals is notified via pre-conditions and post-conditions to link with each other. An agent based approach was introduced by Moon and Thudope in [7], with two agents; (1). Collision avoidance agent(2). Track keeping agent. The track keeping agent keeps the ship on the track which is decided by the instructor and collision avoidance agent applies a rule based collision avoidance to avoid approaching vessels. A fuzzy expert system called Ship Auto-navigation Fuzzy Expert System (SAFES) was proposed in [8] and in [9]. SAFES consist of a collision avoidance system which performs in a multiple ship environment. It is possible to apply SAFES for any waterway configuration such as open sea environment or congested waterway and also possible to apply with any number of ships. A fuzzy control unit contains the instructions and takes the decisions.

Apart from direct target ship navigation mechanisms, some approaches like evolutionary programming[10], [11] is used to develop intelligent navigational systems to assist instructors and these concepts also can be applied with ship handling simulators to avoid collisions between both static and dynamic obstacles.

VI. Conclusions

The main contributions of this research are the use of historic AIS data to obtain intelligent autonomous navigation of surrounding vessels in a marine simulation system and improve the intelligent decision making of the surrounding ship with an ACO algorithm when a collision risk exists. The historical ship tracks can be restored using piecewise cubic Hermite interpolation polynomial with a higher degree of accuracy. Even though, the use of AIS positional data for ship navigation open up huge amount of advantages as discussed in earlier chapters, they have some disadvantages too. The AIS data are hard to collect in deep open sea environments and also not all sea crafts are facilitated with AIS. Therefore, the proposed mechanism is limited into simulating near harbor related sea environments with limited amount of vessels.

An ACO is developed to avoid the collisions when the trainee's ship is in the vicinity of surrounding ships. The radius of the area of observation, apart from being affected by the present encountered situation, would also vary according to different ship type, maneuverability and marine environment conditions. Therefore, a unified range cannot be set for all ships types. The collision avoidance was successful for all head on, crossing and overtaking situations and ACO produces a collision avoided path which complies with the COLREG. Since a grid data structure is used for the ACO algorithm, the generated path tends to have turning points which reduce the smoothness of the path. At the evaluation, it is revealed that the number of obstacles and number of nodes in the environment affects the number of turning points in the collision avoided path. The proposed solution is integrated with the "Vidusayura" ship simulator.

References

[1] R. Hanzu-Pazara, E. Barsan, P. Arsenie, L. Chiotoroiu, and G. Raicu, "Reducing of maritime accidents caused by human factors using simulators in training process," Journal of Maritime Research, vol. 5, no. 1, pp. 3-18, 2008.

[2] B. C. Stewart, "Mounting human entities to control and interact with networked ship entities in a virtual environment," Ph.D. dissertation, Citeseer, 1996.

[3] Y. Shenhua, W. Xinghua, and C. Guoquan, "Design and implement on intelligent ship handling simulator," in Digital Manufacturing and Automation (ICDMA), 2010 International Conference on, vol. 1. IEEE, 2010, pp. 473-477.

[4] I. International Marine Organization. (2013) Preventing collisions. [Online]. Available: http://www.imo.org/OurWork/Safety/navigation/pages/preventing-collisions.aspx

[5] V. R. Group. (2013) Vidusayura. [Online]. Available: www.vidusayura.org

[6] A. Olenderski, M. Nicolescu, and S. J. Louis, "A behavior-based architecture for realistic autonomous ship control," in Computational Intelligence and Games, 2006 IEEESymposium on. IEEE, 2006, pp. 148-155.

[7] J. Moon and D. Tudhope, "An agent-directed marine navigation simulator," Journalof Navigation, vol. 59, no. 3, pp. 461-476, 2006.

[8] K. Hasegawa, J. Fukuto, R. Miyake, and M. Yamazaki, \An intelligent ship handling simulator with automatic collision avoidance function of target ships," RostockWarnemuende, 2012.

[9] K. Hasegawa, "Some recent developments of next generations marine traffic systems," in IFAC Conference on Control Applications in Marine Systems.{2004, Ancona, Italy, 2004, pp. 13-18.

[10] R. Smierzchalski and Z. Michalewicz, "Adaptive modeling of a ship trajectory in collision situations at sea," in Evolutionary Computation Proceedings, 1998. IEEE WorldCongress on Computational Intelligence., The 1998 IEEE International Conference on. IEEE, 1998, pp. 342-347.

[11] R. Smierzchalski, 'Evolutionary-fuzzy system of safe ship steering in a collision situation at sea," in Computational Intelligence for Modelling, Control and Automation,2005 and International Conference onIntelligent Agents, Web Technologies and Internet Commerce, International Conference on, vol. 1. IEEE, 2005, pp. 893-898.

[12] J. R. Irons. (2013) Transverse mercator projection. [Online]. Available: http://landsat.gsfc.nasa.gov/?p=3309