International Journal of Innovative Technology and Exploring Engineering (IJITEE)
ISSN: 2278-3075 (Online), Volume-12 Issue-1, December 2022

Safe Ship Control using Linear Programming

Sebastian Miloch, Wojciech Kińczyk, Mostefa Mohamed-Seghir

Abstract*: The presented work is part of the project to implement a simulator for determining the ship's trajectory in collision situations. The aim of this article is to model an optimal ship control system in collision situations taking into account the International Regulations for the Prevention of Collisions at Sea. The main task was to design and realize a trajectory visualization in the form of a simulation. An analysis of the simulation results was also carried out and used to formulate conclusions. In this paper, an algorithm based on the static linear programming method for determining a multistage ship trajectory was developed and presented. The block diagram of the algorithm and the basics of linear programming are described. A series of simulations in various navigational situations involving a foreign ship sailing on a course of 0°, 45°, 90°, 135°, 180°, 225°, 270°, and 315° was carried out and the influence of the various parameters on the course of the simulation itself was examined. Based on the data obtained, graphs were drawn up to enable an easier analysis of the simulation results. It was also found that, depending on the complexity of the navigational situation, the results are more or less predictable and that, in order to obtain the optimal outcome of the situation, all the simulation parameters must be chosen accordingly.

***Keywords*: Safes Ship Trajectory Ship Control, Collision Situations, Static Optimization, Linear Programming,**

I. INTRODUCTION

Maritime transportation safety is strongly linked to navigation hazards. Nowadays, maritime transportation is one of the safest branches of transportation. Thanks to developing technology, the human factor, which is often the main cause of collisions, is being minimized and humans are being assisted by all sorts of safety systems. Among such solutions is the ARPA (Automatic Radar Plotting Aids) anti-collision system, it has made a huge contribution to shipping safety. This system enables automatic tracking of objects within the range, determination of their movement parameters and support for navigators in making anti-collision maneuver decisions [1] [2] [3] [4] [5] [6].

The task of optimization is to find the best result from a set of acceptable solutions according to the adopted criteria. The goal of static optimization is to find the optimal point, i.e. the point at which the value of the objective is the best. Depending on how the task is formulated, this value will be either the largest or the smallest. The search for such an extreme value can take place in a limited area (local extremum) or in the entire argument space(global extremum) [7] [8] [9] [10] [11]. The purpose of this work is to model a system for optimal ship control in collision situations, design and perform trajectory imaging, simulation and analysis of the results. For the purpose of the program, it was assumed that all encountered vessels move at a constant speed and do not change their initial course. Only its own unit has the ability to change course or speed to avoid collision with another object. At the beginning of the paper, an algorithm for determining the multi-stage optimal trajectory of a vessel based on the static linear programming method was developed and presented. In the next chapter, the construction of the program and how to use it were presented. In the third chapter, a number of simulations in various navigation situations were carried out so that the program's performance could be analyzed. At the end of the paper, all the results of the simulations carried out are summarized in the form of graphs and appropriate conclusions are drawn based on them.

II. PROCES KINEMATIC MODEL

The kinematic model of the safe trajectory of a ship at sea can be represented by an equation of state [12, 13]:

$$
\begin{aligned}
\dot{x} &= V \sin\psi \\
\dot{y} &= V \cos\psi \\
\dot{x}_j &= V_j \sin\psi_j \\
\dot{y}_j &= V_j \cos\psi_j
\end{aligned}
\tag{1}
$$

where:

x - these are the position of the own ship's coordinates,
y - these are the coordinates of the j-th object,
V - own ship speed,
V_j – speed of the j-th object.,
ψ – own ship course,,
ψ_j - course of the j-th object.

These are identified by the automatic radar plotting aids (ARPA) anti-collision system Fig. 1.

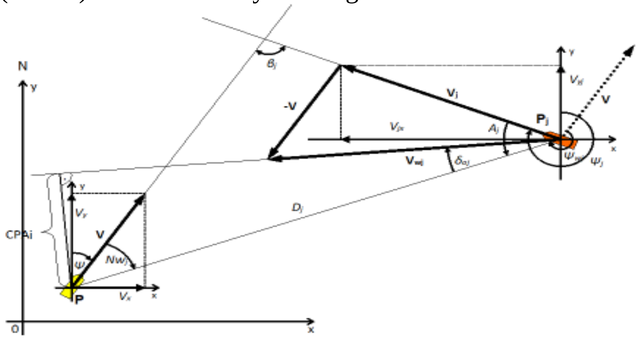

Fig. 1.The situation of own ship passing the jth target presented in a rectangular coordinate system

Manuscript received on 28 November 2022 | Revised Manuscript received on 07 December 2022 | Manuscript Accepted on 15 December 2022 | Manuscript published on 30 December 2022.

***Correspondence Author (s)**
Sebastian Miloch, Department of Ship Automation, Gdynia Maritime University, Gdynia, Poland. Email: 43515@student.umg.edu.pl
Wojciech Kińczyk, Department of Ship Automation, Gdynia Maritime University, Gdynia, Poland. Email: 44715@student.umg.edu.pl
Dr. Mostefa Mohamed-Seghir*, Department of Ship Automation, Gdynia Maritime University, Gdynia, Poland. Email: m.mohamed-seghir@we.umg.edu.pl

Retrieval Number: 100.1/ijitee.A93771212122
DOI: 10.35940/ijitee.A9377.1212122
Journal Website: www.ijitee.org

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)

III. ALGORITHM FOR DETERMINING THE MULTI-STAGE OPTIMAL TRAJECTORY OF A SHIP

The task of optimization is to determine the best (optimal) result of the analyzed problem in terms of a certain criterion (quality index). In the case of determining the optimal trajectory of a ship, this is the quality indicator of optimal control. When determining a ship's own trajectory, it should be taken into account that there are many possible maneuvers. Of the possible maneuvers, the optimal trajectory should be selected in terms of deviation from the set course and the shortest path used to pass the encountered objects, so as to minimize the cost of avoiding a collision.

The algorithm for determining the optimal trajectory is based on several basic points:

1) Input of initial data and their analysis.
2) Checking whether the vessel is on a collision course with another vessel.
3) Determination of linear programming constraints.
4) Calculation of the optimal course.
5) Possible change of the ship's own speed (if speed change is defined).
6) Return to point 2).
7) If a collision does not occur a collision-free course is depicted.
8) Calculation of deviation from the course of own vessel

After entering the parameters of the met objects and own unit, the program analyzing the data calculates the risk of collision. If the calculations indicate the possibility of a collision, the limitations of the linear programming task are defined, while if a collision is not detected, the program completes the calculations and determines the course. If the possibility of a speed change is defined, the program can only reduce the speed, or, if necessary, determine the course as well. Fig. 2 shows the trajectory determination algorithm used in the program.

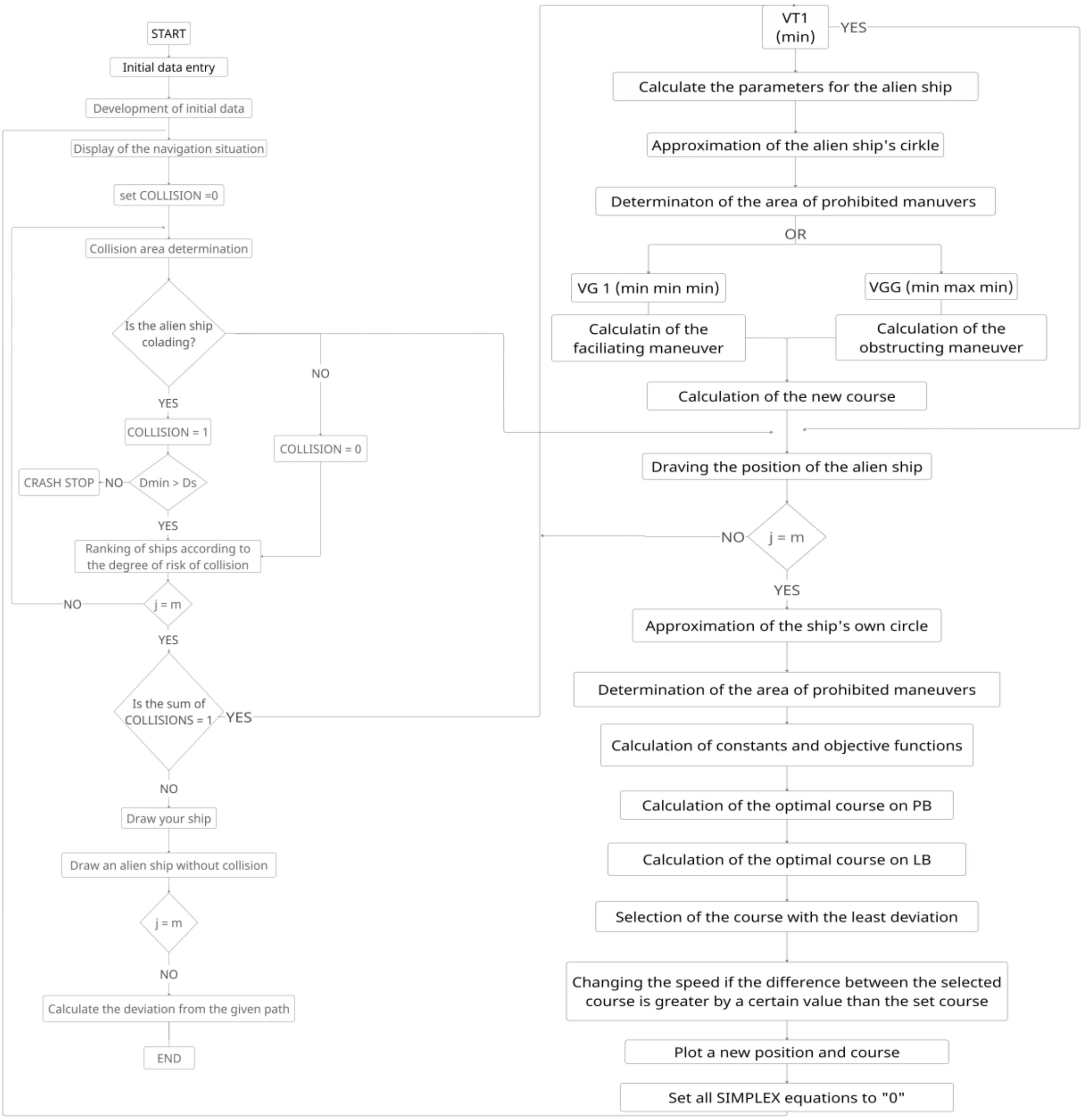

Fig. 2.Algorithm for determining the optimal safe trajectory of a ship

Retrieval Number: 100.1/ijitee.A93771212122
DOI: 10.35940/ijitee.A9377.1212122
Journal Website: www.ijitee.org

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)

A. Linear programming method

The linear programming method was used to optimize the static safe trajectory of the ship. The construction of the used linear optimization function linprog in Matlab R2021 software is as follows [14] [15] [16]:

$$x = lp(f, A, b) \qquad (2)$$

Where:

- f - vector of coefficients of the objective function
- A - matrix of coefficients of inequalities
- b - vector of coefficients of inequality constraints

Linprog thus solves the linear programming problem: min $f'x$ with constraints $A*x<=b$. This function will result in a vector of x with the solution of the linear programming task.

B. Initial data entry

In order to allow the program to correctly calculate the course, it is necessary to enter the relevant parameters of the encountered vessels (in the form of bearing, distance, course and speed) and the own vessel (speed and set course), as well as the relevant parameters of the simulation being carried out. The position of the own vessel is set permanently so that the vessel starts from the point (0,0) in the coordinate system.

It should be remembered that this is a simulation and, in reality, data about the encountered vessels and the parameters of the own ship arrive at the ARPA systems from radar, log, GPS and other devices monitoring the ship and the area around it.

C. Designation of the collision area

The variable "collision" was used to determine the collision course, which takes the value 1 if the ship is on a collision course or 0 if the course is collision-free. The variable must be reset to zero each time before proceeding with calculations, since it takes on a different value in each collision check step. This area must be determined for each of the encountered vessels, making it possible to present the navigational situation of our own vessel and check whether the selected course of our vessel is collision, and whether any of the encountered vessels are on a collision course with own ship.

D. Calculating the optimal rate

The program, after approximating the ship's own circle and determining the area of forbidden maneuvers, defining limitations and target functions, determines the starboard course and then calculates the difference from the set course in degrees. Exactly the same calculations are also performed for the set course to the port side. With these calculations, the program can determine which maneuver will be the most optimal in terms of deviation from the set course. Each time the appropriate course is calculated and applied, its value will be displayed in the simulation window as the current course. Throughout the simulation process, the program user can see how the current course changes relative to the set course.

E. Changing the speed of your own ship

If you define in the simulation parameters window "Deviation from course, at which to decrease speed" to a value below 361°, the program, if your own ship takes a course above the entered value, will begin to reduce its speed by the percentage you have specified for the "Percentage of speed reduction" window (set to 25% by default). The speed will be reduced gradually until your own ship finishes maneuvering or obtains a deviation value smaller than the one entered in the "Deviation from course at which to reduce speed" window. The speed at which the ship is moving will be displayed in the " Current Speed" window throughout the simulation. Once your own ship has exited the collision situation and returned to the preset course, the speed at which the ship has completed its maneuvers and from which it can begin to increase its speed to the preset speed will be displayed in this window.

IV. SIMULATION

The simulations were performed for different navigational situations. In each of the different simulations, the parameters of the encountered vessels change, such as the number of vessels encountered, their bearing, distance, speed and course, as well as the parameters of the own vessel and the simulation itself. In depicting each situation, all the trajectories of the own ship's movement in a given scenario were superimposed with the changing parameter of the safe distance, so as to see the effect of changes in this distance on the deviation from the set course. Table - I assigns the values of the safe distance to the numbering shown located on the each trajectory.

Table - I: Numbering of individual trajectories by safe distance

No.	Safe distance
a	0.5
b	1.0
c	1.5
d	2.0
e	2.5
f	3.0
g	3.5
h	4.0
i	4.5
j	5.0
k	5.5
l	6.0

A. Situation with a foreign ship sailing on a 90° course

Table - II lists the initial parameters for all simulations of this situation. Fig. 3 shows the trajectories of the own ship's movement for different values of the safe distance.

Table - II: Simulation parameters with a foreign ship sailing on a 90° course

Parameter	Unit	Value
Speed of own ship	[kn]	20
Course of own ship	[°]	0
Distance of the encountered craft	[NM]	4
Bearing of the encountered craft	[°]	0
Course of the encountered craft	[°]	90
Speed of the encountered craft	[kn]	16
Maneuvering time	[min]	3
Advance time	[min]	0
Deviation from course at which to reduce speed	[°]	361
Percentage of speed reduction	[%]	25

Retrieval Number: 100.1/ijitee.A93771212122
DOI: 10.35940/ijitee.A9377.1212122
Journal Website: www.ijitee.org

Published By:
Blue Eyes Intelligence Engineering
and Sciences Publication (BEIESP)

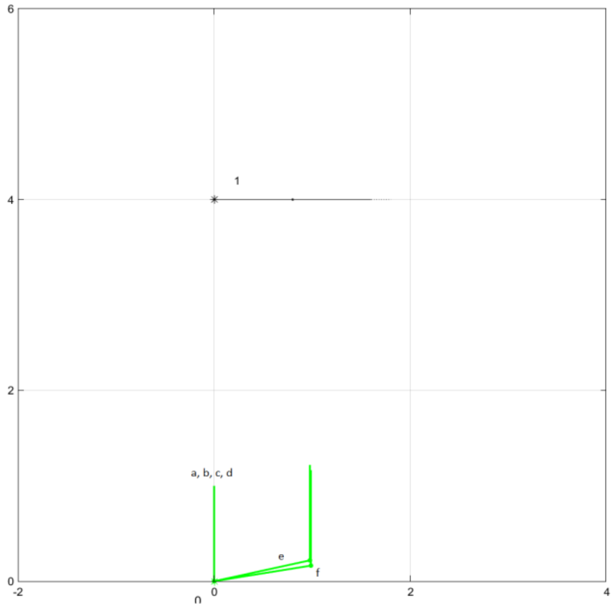

Fig. 3.Trajectory of ship movement - foreign ship on a 90° course

B. Situation with a foreign ship sailing on a 135° course

Table - III lists the initial parameters for all simulations of this situation. Fig. 4 shows the trajectories of the own ship's movement for different values of the safe distance.

Table - III: Simulation parameters with a foreign ship sailing on a 135° course

Parameter	Unit	Value
Speed of own ship	[kn]	20
Course of own ship	[°]	0
Distance of the encountered craft	[NM]	4
Bearing of the encountered craft	[°]	0
Course of the encountered craft	[°]	135
Speed of the encountered craft	[kn]	16
Maneuvering time	[min]	3
Advance time	[min]	0
Deviation from course at which to reduce speed	[°]	361
Percentage of speed reduction	[%]	25

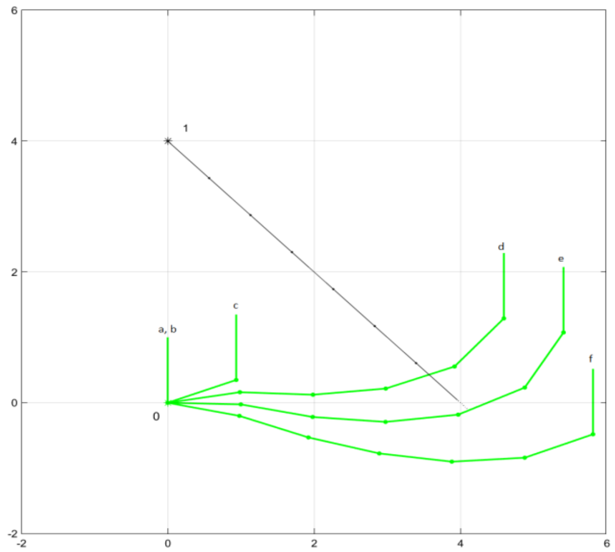

Fig. 4.Trajectory of ship movement - foreign ship on 135° course

C. Scenario with 30 ships encountered

Table - IV lists the initial parameters for all simulations of this situation. Fig. 5 shows the trajectories of the own ship's movement for different values of the safe distance.

Table - IV: Simulation parameters with 30 ships encountered

Parameter	Unit	Value
Speed of own ship	[kn]	11
Course of own ship	[°]	23
Maneuvering time	[min]	3
Advance time	[min]	0
Deviation from course at which to reduce speed	[°]	90
Percentage of speed reduction	[%]	25

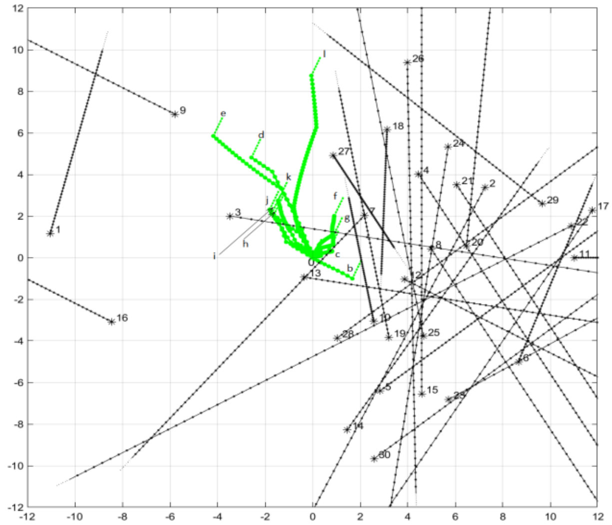

Fig. 5.Movement trajectory of 30 ships encountered

V. ANALYSIS OF SIMULATION RESULTS

The parameter that was changed in each scenario was the safe distance, in order to be able to summarize the effect of this value on the course of the simulation. For situations that are less complex in terms of the number of units, from the plotted trajectories alone, one can see the relationship of increasing the deviation from the course while increasing the safe distance. However, for more complex situations in terms of the number of units encountered, and with the speed reduction option activated, the situation already looks less predictable.

A. The results of the situation of the meeting of one unit

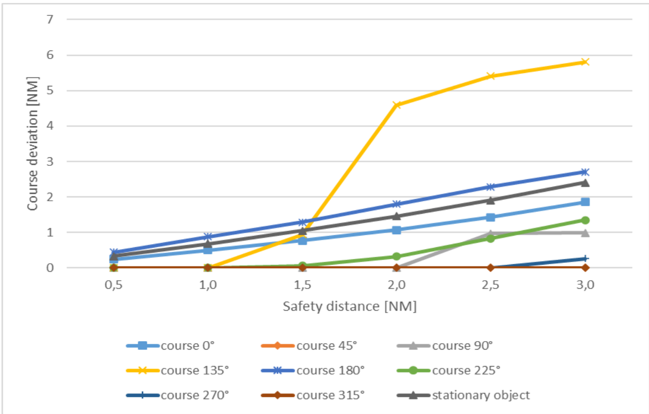

Fig. 6.Diagram of the dependence of the deviation from the safe distance for the situation of meeting one unit

Retrieval Number: 100.1/ijitee.A93771212122
DOI: 10.35940/ijitee.A9377.1212122
Journal Website: www.ijitee.org

Published By:
Blue Eyes Intelligence Engineering and Sciences Publication (BEIESP)

Fig. 6 compares the deviation from the course in relation to the safe distance of a craft located 4 nautical miles ahead of the bow of its own vessel for its different courses and for the situation when the object remains stationary.

The largest deviation from the set course was recorded during the situation in which the encountered vessel took a course of 135°. This is due to the fact that the own vessel intended to avoid the encountered craft from the right side, and this led to a large deviation from the set course, relative to the other situations.

For courses of 45° and 315°, the encountered object was not a threat, which did not lead to any deviation, and the own ship could ignore the craft.

B. Results of the situation of encountering multiple ships

In this scenario, the speed reduction function led to the opposite result. Fig. 7 shows how reducing speed for safe distances of 3 to 5 NM resulted in a significant increase in deviation from course. From this example, it can be concluded that when there is a high density of vessels, reducing speed can worsen the outcome of evasive maneuvers.

Along the lines of the simulation result of the scenario with 20 vessels, in the scenario with 30 vessels, there is also a significant increase in the deviation from the course in the specific safe distance range.

Looking at scenario with 50 ships on Fig. 7 , it can be seen that the deviation increases, up to a value of 2.5 NM safe distance. The deviation value then decreases significantly. This situation is due to the activation of the speed reduction process, which in response led to a positive response in the form of reduced deviation.

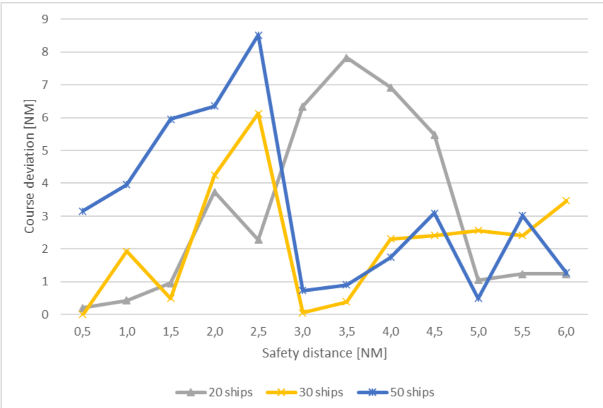

Fig. 7.Dependence chart of deviation from safe distance for a scenario with multiple ships

C. The results of the situation of meeting one unit in three different directions

A summary of the three test scenarios is included in Fig. 12. The value of the deviation increases in these cases along with the safe distance, and no significant deviations from the trend appear here. The differences in the deviations of the individual scenarios are due to the position of the units encountered and their course.

The largest deviations appeared in the scenario when the encountered vessel sailed on a course opposite to ours, but its position was shifted to the right relative to its own vessel. The own vessel, in accordance with the rules of the International Regulations for Preventing Collisions at Sea, sought to avoid the encountered object from the right. As a result of this maneuver, the ship had to deviate from the set course more significantly than in the other situations.

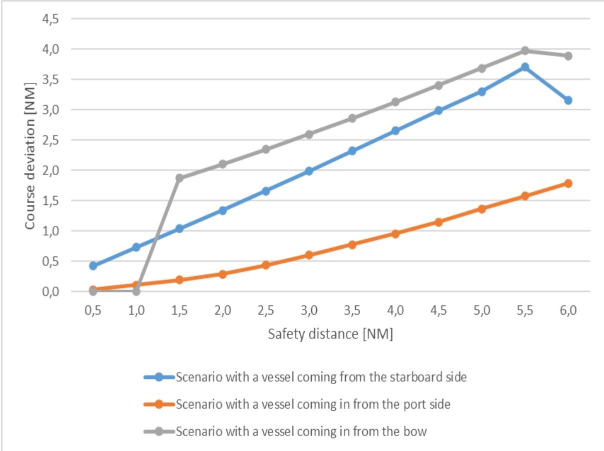

Fig. 8.Graph of the dependence of the deviation from the safe distance for scenarios with a unit flowing from 3 different directions in succession

VI. CONCLUSION

Analyzing all the above simulations, it can be seen that the most predictable situations are the less complex ones, when there is not much density of encountered vessels. In this case, selecting the appropriate parameters to achieve the most favorable result is much easier than in more complex navigation situations.

In order to bring about the lowest possible deviation from the course, it is necessary to select the value of the deviation at which to start reducing speed so as not to lead to a situation worse than if this option is turned off. When designing a situation, changing any parameter can lead to a completely different simulation result.

It should be noted that the change of the ship's course in real conditions takes place over a much longer period of time, and the program presented here is only a simulator, which does not take into account many factors such as the size of the vessels or the limitations of the body of water in the form of islands or coastline, for example. The safe distance parameter in the real situation depends on the current meteorological situation. During poor visibility, the safe distance will be significantly higher than in good visibility conditions.

The designed simulator can be a valuable tool for teaching purposes to demonstrate the simplified principle of the ARPA system, which tracks multiple objects within range of radar systems in real time. The simulator also demonstrates the use of the linear programming method and how it works.

Additional features of this type of work would be the ability to introduce body of water restrictions such as islands and coastlines, define the size of all units, or introduce a depth parameter. A useful feature would also be the introduction of control of the courses of units encountered by one's own ship, so that not only one's own ship would calculate a collision-avoiding trajectory, but all units added to the simulation.

ACKNOWLEDGMENT

Funding: The project is financed in the framework of the program by the Ministry of Science and Higher Education called "Regionalna Inicjatywa Doskonałości" in the years 2019–2022 (project number: 006/RID/2018/19; the sum of financing: 11,870,000 PLN).

REFERENCES

1. R. Hedjar and M. Bounkhel, 'An automatic collision avoidance algorithm for multiple marine surface vehicles', Int. J. Appl. Math. Comput. Sci., vol. 29, no. 4, 2019. [CrossRef]
2. Y. Huang, L. Chen, R. R. Negenborn, and P. Van Gelder, 'A ship collision avoidance system for human-machine cooperation during collision avoidance', Ocean Eng., vol. 217, p. 107913, 2020. [CrossRef]
3. A. Lazarowska, 'A discrete artificial potential field for ship trajectory planning', J. Navig., vol. 73, no. 1, pp. 233–251, 2020. [CrossRef]
4. J. Lisowski, 'Sensitivity of Safe Trajectory in a Game Environment on Inaccuracy of Radar Data in Autonomous Navigation', Sensors, vol. 19, no. 8, p. 1816, Apr. 2019, doi: 10.3390/s19081816. [CrossRef]
5. M. Mohamed-Seghir, 'Methods based on fuzzy sets to solve problems of Safe Ship control', in Novel Algorithms and Techniques in Telecommunications and Networking, Berlin, 2010, pp. 373–377. doi: 10.1007/978-90-481-3662-9_64. [CrossRef]
6. R. Zaccone and M. Martelli, 'A collision avoidance algorithm for ship guidance applications', J. Mar. Eng. Technol., vol. 19, no. sup1, pp. 62–75, 2020. [CrossRef]
7. A. Witkowska, R. Smierzchalski, and P. Wilczynski, 'Trajectory planning for Service Ship during emergency STS transfer operation', in Trends in Advanced Intelligent Control, Optimization and Automation, Cham, 2017, vol. 577, pp. 514–523. doi: 10.1007/978-3-319-60699-6_50. [CrossRef]
8. D. Zhang, 'Optimization Design of Ship Rudder Steering Stability Optimization Control System: Modelling and Computer Simulations', in 2021 5th International Conference on Electronics, Communication and Aerospace Technology (ICECA), 2021, pp. 984–987. [CrossRef]
9. G. Xia, Z. Han, B. Zhao, and X. Wang, 'Local path planning for unmanned surface vehicle collision avoidance based on modified quantum particle swarm optimization', Complexity, vol. 2020, 2020. [CrossRef]
10. R. Fıskın, H. Kisi, and E. Nasibov, 'A research on techniques, models and methods proposed for ship collision avoidance path planning problem', Int. J. Marit. Eng., vol. 160, no. A2, 2018. [CrossRef]
11. S. Guo, X. Zhang, Y. Du, Y. Zheng, and Z. Cao, 'Path planning of coastal ships based on optimized DQN reward function', J. Mar. Sci. Eng., vol. 9, no. 2, p. 210, 2021. [CrossRef]
12. A. Lazarowska, 'Comparison of discrete artificial potential field algorithm and wave-front algorithm for autonomous ship trajectory planning', IEEE Access, vol. 8, pp. 221013–221026, 2020. [CrossRef]
13. J. Lisowski, 'Computational Intelligence in Marine Control Engineering Education', Pol. Marit. Res., vol. 28, no. 1, pp. 163–172, Mar. 2021, doi: 10.2478/pomr-2021-0015. [CrossRef]
14. D. Zhuge, S. Wang, and D. Z. Wang, 'A joint liner ship path, speed and deployment problem under emission reduction measures', Transp. Res. Part B Methodol., vol. 144, pp. 155–173, 2021. [CrossRef]
15. F. Nurprihatin, T. Regina, and G. D. Rembulan, 'Optimizing rice distribution routes in Indonesia using a two-step linear programming considering logistics costs', in Journal of Physics: Conference Series, 2021, vol. 1811, no. 1, p. 012010. [CrossRef]
16. A. Kim, H. Kim, H. Lee, B. Lee, and H. Lim, 'Comparative economic optimization for an overseas hydrogen supply chain using mixed-integer linear programming', ACS Sustain. Chem. Eng., vol. 9, no. 42, pp. 14249–14262, 2021. [CrossRef]

AUTHORS PROFILE

Sebastian Miloch Completed engineering studies at the Faculty of Electrical Engineering, majoring in Electrical Engineering, specializing in Computer Control Systems. Currently in the course of master's studies at the Gdynia Maritime University. I am interested in optimization methods, broadly understood automation, artificial intelligence in the use of the land and sea industry. For my work, I use Matlab/Simulink, Tia Portal, and C++ and Python development environments.

Wojciech Kińczyk Completed engineering studies at the Faculty of Electrical Engineering, majoring in Electrical Engineering, specializing in Computer Control Systems. Currently in the course of master's studies at the Gdynia Maritime University. I am interested in optimization methods, broadly understood automation, artificial intelligence in the use of the land and sea industry. For my work, I use Matlab/Simulink, Tia Portal, and C++ and Python development environments.

Dr. Mostefa Mohamed-Seghir, Department of Ship Automation, Faculty of Electrical Engineering, Gdynia Maritime University, Poland, **Interests**: Fuzzy logic; control engineering; machine learning; optimization; model predictive control; power electronics.
E-mail: m.mohamed-seghir@we.umg.edu.pl
Website: https://we.umg.edu.pl/kao/pracownicy