COMPUTATIONAL INTELLIGENCE IN MARINE CONTROL ENGINEERING EDUCATION

Józef Lisowski
Gdynia Maritime University, Poland

ABSTRACT

This paper presents a new approach to the existing training of marine control engineering professionals using artificial intelligence. We use optimisation strategies, neural networks and game theory to support optimal, safe ship control by applying the latest scientific achievements to the current process of educating students as future marine officers. Recent advancements in shipbuilding, equipment for robotised ships, the high quality of shipboard game plans, the cost of overhauling, dependability, the fixing of the shipboard equipment and the requesting of the safe shipping and environmental protection, requires constant information on recent equipment and programming for computational intelligence by marine officers. We carry out an analysis to determine which methods of artificial intelligence can allow us to eliminate human subjectivity and uncertainty from real navigational situations involving manoeuvring decisions made by marine officers. Trainees learn by using computer simulation methods to calculate the optimal safe traverse of the ship in the event of a possible collision with other ships, which are mapped using neural networks that take into consideration the subjectivity of the navigator. The game-optimal safe trajectory for the ship also considers the uncertainty in the navigational situation, which is measured in terms of the risk of collision. The use of artificial intelligence methods in the final stage of training on ship automation can improve the practical education of marine officers and allow for safer and more effective ship operation.

Keywords: ship control,marine engineering curriculum,computational intelligence,game theory,computer simulation

INTRODUCTION

This paper addresses the theoretical and practical training of students as future ship officers in the field of ship control engineering and its functional facilities and processes. Control engineering includes the fundamentals of automation and specialist subjects in the field of automation, computer science, electronics, optimisation and artificial intelligence (AI). Knowledge in the field of control engineering is acquired successively, through the following stages: lectures, laboratory work, simulations and operational practice on a ship.

In laboratory sessions dealing with the fundamentals of automation, students become acquainted with the two main decision support tools of AI and game theory. These are represented by two exemplary control algorithms, which allow for a better understanding and application of training in simulator sessions.

The purpose of control engineering training within marine territories is to pass on information on the development of ship computerisation frameworks and associated tasks, as set out by the International Maritime Organisation in the STCW-95 convention. The training of marine officers on automation should include the modern equipment carried on ships, in addition to control engineering theory and modern control engineering techniques, and may follow the textbook by Nise [19]. Heiselberg and Stateczny [7], Huang et al. [10], Lazarowska [13], Lebkowski [14] and Zhuang et al. [32] have shown that computer-aided navigation training on safe ship

controls that involves computational intelligence methods and game theory has become important. Annual statistical studies prepared by the European Maritime Safety Agency (EMSA), have demonstrated that around 80% of maritime accidents are caused by the subjectivity of a navigator when assessing the navigation situation, and before making the final manoeuvring decision.

An analysis by Ahn et al. [1] has shown that the use of AI methods in the form of a neural network to decide on the correct direction for the ship during situations of excessive proximity to others allows for the consideration of the navigator's subjectivity during the impact on the final safe manoeuvre.

Liu et al. [16] demonstrated that the use of game hypothesis to decide on a safe direction for a ship in a collision event allows for the consideration of uncertainty when encountering large numbers of nearby ships, especially in areas of limited visibility. In both cases, there are many solutions for determining safe directions for ships, and the best of these solutions should be selected, i.e. the optimal solution based on a particular criterion, such as extending the cruise route to ensure safe avoidance of other ships. As reported by Guenin et al. [5], Speyer et al. [24], Szlapczynska et al. [25], Witkowska et al. [30] and Yong [31], both static and dynamic optimisation methods can be used.

The aim of this research is to present two control algorithms to represent the two main components that are important when making decisions in simulator training: the subjectivity of decisions, which can be described by an artificial neural network model, and the uncertainty of navigation situations, as measured by the possible risk of collision, which can be mapped using a model of an appropriate game. These studies can be very valuable both when designing of new versions of simulators and when using them for training.

These new elements link the Bellman optimality principle with a neural network model to generate domains for ships and a matrix game, with collisional risk to the synthesis of safe steering of the ship's movement.

CONTROL ENGINEERING

Control engineering is taught within the Faculties of Navigation, Marine Engineering and Electrical Engineering at the Maritime University. Cwilewicz et al. [4] found that the curriculum grades within each department were different, and addressed both the explicit responsibilities of graduates when on board and the requirements arising from maritime conventions.

Cadets in the Faculties of Navigation and Marine Engineering mainly receive instruction in the single subject of automation fundamentals, while those in the Faculty of Electrical Engineering receive a broader range of training in automation due to their wider range of official duties when operating devices onboard. Based on the principles identified by Borrego et al. [3], Henri et al. [8] and Lattuca et al. [12], it is possible to design curricula in such a way that cadets receive

practical instruction in automatic control techniques, which is more useful within the profession of ship electroautomatics. The two-level system for training in automation within the Faculty of Electrical Engineering includes the following topics, delivered via lectures and laboratories:

- At the basic engineering level:
  Basics of Automatics
  Electronics and Power Electronics
  Digital Technology
  Mechatronics and Robotics
  Devices and Control Systems Engineering
  Automated Electrical Ship Power Drives
  Automation of Ship Power Systems
  Programmable Logic Controllers
  Visualisation of Control Processes
- At the more advanced Master's level:
  Advanced Control Engineering Methods
  Optimisation Methods
  AI Methods
  Digital Control Systems
  Automation of Electrical Power Plants
  Distributed Control Systems
  Computer Control Support.

FUNDAMENTALS OF AUTOMATION

The course on the Basics of Automatics includes the following themes:
1) Theory and techniques of automation:
   - Principles of automation-basic definitions, open and closed-loop feedback control systems, types of automatic control;
   - Methods of describing the static and dynamic properties of the physical elements of control systems: transfer function, time and frequency responses, state equations;
   - Basic physical elements of control systems and their features;
   - Characteristics of typical industrial control objects;
   - Identification of control objects;
   - Structural block diagrams of automatic control systems;
   - The requirements that shoud be met by automatic control systems-stability margins, quality control within transitory states, accepted steady-state errors, correction of control systems;
   - PID controllers-structures and characteristics, selection of optimal settings;
   - Direct digital control;
   - Complex systems of automatic control-cascade control, closed-open systems, multivariable control systems;
   - Nonlinear systems, relay control, two-set controllers, three-set controllers, step controllers.
2) Modern control systems:
   - Extremal control;
   - Optimal control;
   - Adaptive control;
   - Game control;

- Computational intelligence control-expert systems, fuzzy control, artificial neural networks, evolutionary algorithms.
3) Systems engineering for control of ship movement:
  - Methods for describing the dynamics of a ship as an automatically controlled object;
  - Ship control systems-maintenance of course and trajectory, control of the ship's speed, dynamic positioning, precise ship steering via thrusters, roll stabilisation, safe control in collision situations, optimisation of the ship's route.
4) Shipboard control systems:
  - Main propulsion system of a ship with adjustable pitch propeller;
  - Power generation plant;
  - Cargo refrigerated hold;
  - Ballast and bilge systems;
  - Thruster system;
  - Fire detection, alarm and fighting system;
  - Microprocessor system for monitoring and control of the engine room.

PRACTICAL EDUCATION

Trussell et al. [28] and Weisner et al. [29] report that the teaching process takes place in the following three forms:
- Lectures, exercises and laboratory exercises;
- Operating experience through a school (seafaring and manoeuvring practices) and on board commercial ships (operational practices);
- Exercises on training simulators (navigational, radar, cargo and specialist).

Based on experience, Guzey et al. [6], Nikolic [18] and Trevelyan [27], from the perspective of the forthcoming work of a graduate as an administrator of control frameworks onboard a ship, the training on ships and test system practices play a key role.

COMPUTER SUPPORT FOR OPTIMAL AND SAFE SHIP CONTROL

Safe ship control relies upon constant monitoring of the conditions at sea, anti-collision manoeuvres, its acknowledgement and the safe control to the closest return point, recently assigned on the electronic map. These factors are crucial when deciding on a safe trajectory for a ship, since a ship's single-manoeuvre system or potential speed is a multi-stage decision-making process, as discussed by Bellman [2].

The implementation of a multi-stage safety control system is difficult, due to the complex properties of the control process. In training strategies that involve the selection of a manoeuvre or trajectory, one should expect the control algorithms programmed within the microprocessor controller, which defines the anti-collision framework of the ARPA radar, as illustrated in Figure 1.
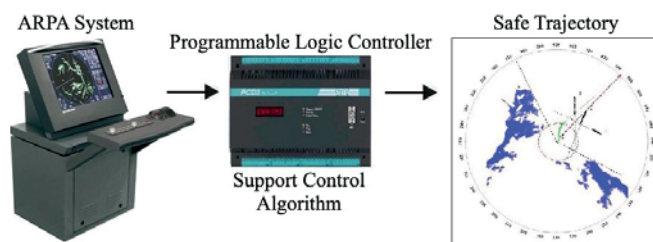

Fig. 1. Computer navigator decision support system in a collision situation at sea.

Ship control relies on the accuracy of traffic data, and uses mathematical descriptions of control processes. These descriptions consist of three components: kinematic and dynamic equations for the motion of the ship, sea waves, and complex navigational situations, as determined by the number of ships passed and the visibility at sea.

Kula [11], Liu [17] and Nise [19] show how a wide variety of models have different impacts on the synthesis of various control algorithms and the impact of safe ship motion controls.

The aim of this article is to show that it is possible to determine the single best solution from the many possible alternatives for calculating the safe trajectory of a ship, i.e. the one that ensures the least loss of the way to the safe passing of the encountered ships.

COMPUTER SIMULATION OF A NEURO-OPTIMAL SAFE TRAJECTORY THAT TAKES ACCOUNT OF NAVIGATOR SUBJECTIVITY

Over 80% of ship collisions are caused by human factors arising from subjective assessments of navigational situations and manoeuvring decisions. It is estimated that about half of these losses could be avoided by using computer programmes to support manoeuvring decisions by the navigator, based on AI, game theory and optimisation methods. When educating students as future officers for sea-going vessels, computer simulations or programmes should be included that take into consideration both the subjectivity of the navigator when making the final manoeuvring decision and the characteristics of the real and often complex navigational situation at sea.

The basic criterion used to ensure the quality of ship steering is the safe movement of ships within a given area, and this is considered by the simulation algorithm in the form of limitations on the state of the steering process. Optimisation of this control task is achieved by minimising the changes needed to the cruise route in order to safely pass all encountered ships. Since most anti-collision manoeuvres are performed in practice by changing the course while maintaining a constant speed, this task is reduced to one of time-optimal control.

Hongguang and Yong [9] illustrate the danger of ship collisions, and show that it is possible to assign certain areas to each ship in the form of domains. These domains may have fixed or variable shapes depending on the collisional

risk, and an artificial neural network implemented in Matlab software can be used to generate these. Figure 2 shows four types of ship domains, for which values are assigned based on the dimensions and speed of the ship and the safe distance at which other ships should be passed under real visibility conditions at sea.

As described in [9], the domains used for the ships may be in the form of a circle, a hexagon, an ellipse or a parabola; the choice depends on the relative speed of other passing ships, and can be changed based on the responses from an appropriately designed neural network that assesses the level of collision risk.
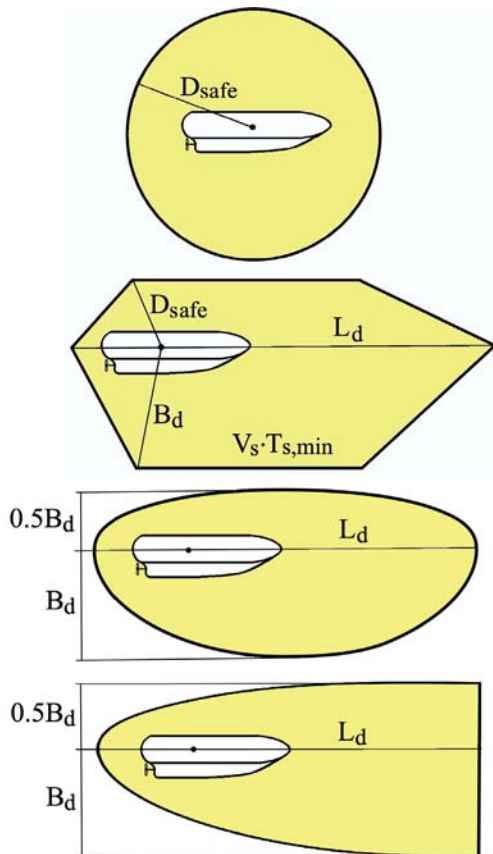

Fig. 2. Shapes of neural encountered ships domains: D_{safe} – safe distance, B_d – dynamic length of the ship, L_d – dynamic beam of the ship, V_s –speed of met s ship, $T_{s,min}$ - minimum time to approach with s met ship.

The neural network is characterised by six input quantities *u* that describe the current collision situation, and which are gathered from measuring devices such as radar, logs, and gyrocompasses. These are combined to form a single output quantity *r* that represents the risk of collision:

$$\boldsymbol{r} = A[W\boldsymbol{u}] \quad \text{(1)}$$

$$\boldsymbol{u} = \left[D_j, N_j, V_j, \psi_j, V, \psi\right] \quad \text{(2)}$$

$$\boldsymbol{r} = \begin{cases} 0.1 & \rightarrow safe\ situation \\ 0.3 & \rightarrow attention \\ 0.5 & \rightarrow collision\ risk \\ 0.7 & \rightarrow dangerous\ situation \\ 0.9 & \rightarrow collision \end{cases} \quad \text{(3)}$$

This leads to the following equation:

$$I^* = \min_A \Sigma(r_i - r_{ei})^2 \quad \text{(4)}$$

where *A* are the activation functions of the neural network layers; r_i and r_{ei} are the real and expected network responses; Σ is a mathematical measurement of the neural network learning processes; and *i* is the time parameter.

Figure 3 shows the structure of a neural network with three layers: the input and hidden layers have activation functions with a digressional nature and the output layers have sigmoidal activation functions.
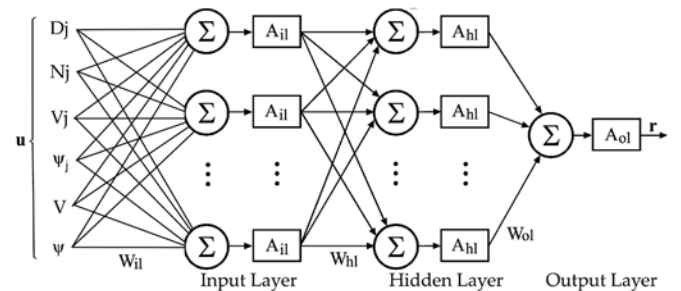

Fig. 3. Neural network for generating ship domains: W_{il} –weight factors of the input layer; W_{hl} –weight factors of the hidden layer; W_{ol} –weight factors of the output layer; A_{il} –activation functions of the input layer; A_{hl} –activation functions of the hidden layer; A_{ol} –activation function of the output layer.

When mapping current navigational situations using a neural network, a backpropagation error algorithm with comprehensive learning and moment indicators is used. The data needed for this learning process were prepared using an ARPA system simulator by 285 navigators.

The ship's dynamics can be described by the state equations in discrete form:

$$\begin{cases} \dot{x}_{i+1}^0 = x_i^0 + V_i\,\Delta t_{i+1}\,sin\,\psi_i \\ \dot{y}_{i+1}^0 = y_i^0 + V_i\,\Delta t_{i+1}\,cos\,\psi_i \\ \dot{\psi}_{i+1} = \psi_i + \dot{\psi}_i\,\Delta t_{i+1} \\ \ddot{\psi}_{i+1} = \dot{\psi}_i + \frac{1}{T_1}\left(-\dot{\psi}_i - k_1\,\dot{\psi}_i^2 + k_2\,\alpha_i\right)\Delta t_{i+1} \\ \dot{V}_{i+1} = V_i + \dot{V}_i\,\Delta t_{i+1} \\ \ddot{V}_{i+1} = \dot{V}_i + \frac{1}{T_2 T_3}\left[-(T_2 + T_3)\dot{V}_i - V_i + k_3 n_i\right]\Delta t_{i+1} \end{cases} \quad \text{(5)}$$

where:

(x^0, y^0) are the coordinates of the ship's position, ψ is the ship's course, $\dot{\psi}$ is the angular turning speed, $\ddot{\psi}$ is the angular acceleration of the ship, V is the ship's speed, $\dot{V}$ is its acceleration, $\ddot{V}$ is the change in the acceleration, α is

the rudder angle, n is the rotational speed of the screw propeller, (k_1, k_2, k_3) are proportionality parameters, and (T_1, T_2, T_3) are the time parameters of the dynamic model.

Determining the optimal route for the ship can be treated as a multi-stage decision-making process, and can be solved using Bellman's dynamic programming method. We use a quality criterion Q, which represents the smallest extension to the voyage route that is necessary to safely pass encountered ships, leading to time-optimal steering while matching the speed of the other ships:

$$Q(\alpha, n) = \int_0^{t_K} V dt \cong V \int_0^{t_K} dt = Q_{min} \qquad (6)$$

Bellman's method defines the principle of optimality, stating the optimal strategy characterises that whatever the initial state or steering may be, the remaining controls must form the optimal strategy from the point of view of the state resulting from the first control:

$$Q_{min,k} = \left\{ \begin{array}{l} Q_{min,k-1}\left[x_k^0, y_k^0, \psi_{k-1}, \dot{\psi}_{k-1}, V_{k-1}, \dot{V}_{k-1}\right] + \\[2mm] \Delta t_k \left[\begin{array}{l} x_k^0, y_k^0, x_{k+1}^0 \left(\begin{array}{l} x_k^0, \psi_k\left(\begin{pmatrix} \psi_{k-1}, \dot{\psi}_{k-1} \\ \begin{pmatrix} \dot{\psi}_{k-2}, \\ \alpha_{k-2}, \Delta t_{k-2} \end{pmatrix}, \Delta t_{k-1} \end{pmatrix} \right), \\ V_k\left(V_{k-1}, \dot{V}_{k-1}, \begin{pmatrix} \dot{V}_{k-2}, \\ n_{k-2}, \Delta t_{k-2} \end{pmatrix}, \Delta t_{k-1} \right) \end{array} \right), \\[6mm] y_{k+1}^0 \left(y_k^0, \psi_k\left(\begin{array}{l} \psi_{k-1}, \dot{\psi}_{k-1}(\psi_{k-2}, \alpha_{k-2}, \Delta t_{k-2}, \Delta t_{k-1}) \\ V_k(V_{k-1}, \dot{V}_{k-1}(\dot{V}_{k-2}, n_{k-2}, \Delta t_{k-2}), \Delta t_{k-1}) \end{array} \right), \\ V_k(V_{k-1}, \dot{V}_{k-1}(\dot{V}_{k-2}, n_{k-2}, \Delta t_{k-2}), \Delta t_{k-1}) \end{array} \right) \end{array} \right] \end{array} \right\}$$
$$k = 3, 4, \dots, K \qquad (7)$$

The time required to reach the k-th stage can be determined as follows:

$$\Delta t_k = \frac{\Delta s_k(x_k^0, y_k^0, x_{k+1}^0, y_{k+1}^0)}{V_k} \qquad (8)$$

It is results from this how calculations utilising this strategy are typically started from the last stage, and afterwards the procedure goes towards the first. According to Bellman's theorem [2], the collision avoidance procedure meets the conditions of duality, and according to this principle, the optimal trajectory for a ship in a collision situation can also be determined using the optimisation principle, starting with the calculation of the first stage before coordinated towards the last stage, as illustrated in Figure 4.
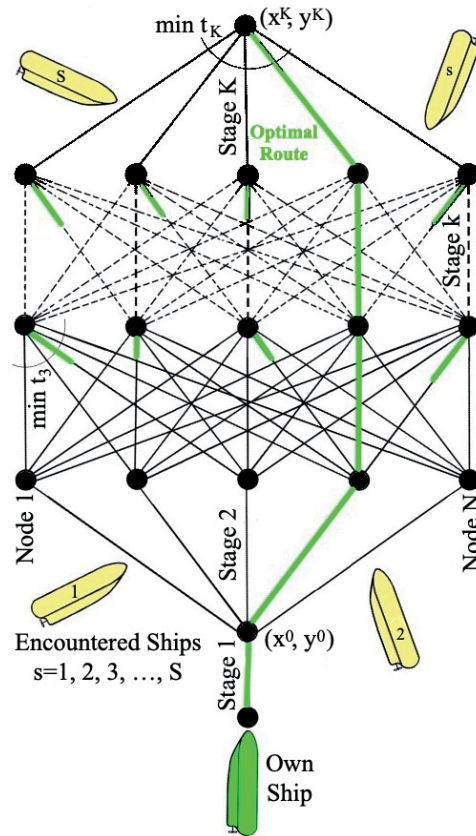

Fig. 4. Multi-stage decision-making process involving dynamic programming of the optimal voyage route, in which the domains of passed ships are mapped.

The navigation situation in Kattegat Strait is shown in Figure 5, and was used in simulation tests of the safe ship control algorithm, as recorded in the ARPA anti-collision system and installed on the research-training vessel.
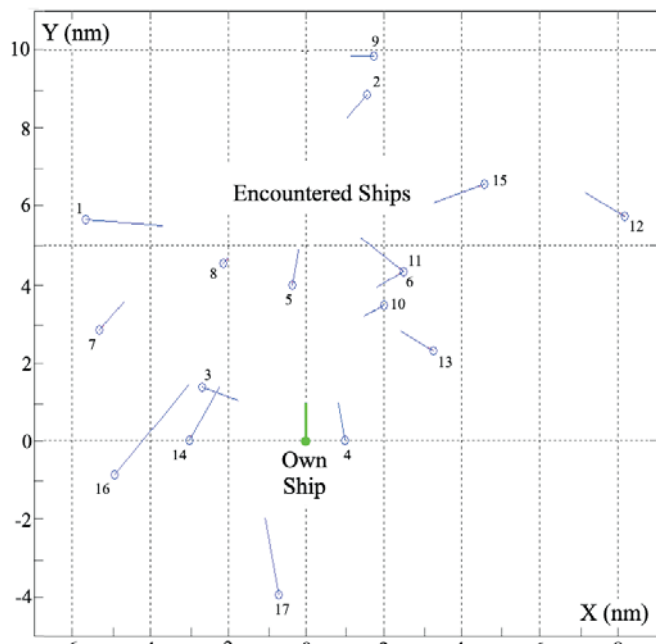

Fig. 5. Navigational situation affecting the movement of a ship between seventeen passing vessels.

Figures 6 and 7 show the neuro-optimal safe trajectories for ships under conditions of good and restricted visibility, which were processed using a programme implemented in Matlab.
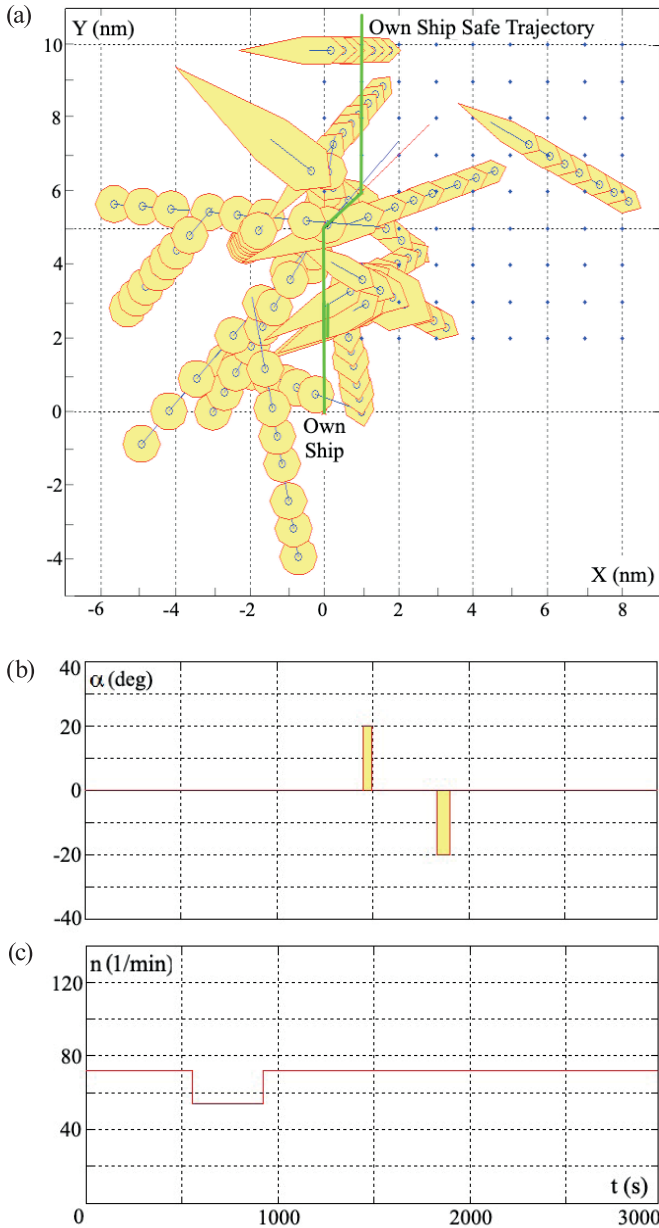

*Figure 6. Results of a computer simulation of the neuro-optimal safety of a ship's trajectory and control sequence under good visibility conditions at sea for $D_s = 0.5$ nm, with circular and **hexagonal** domains, $Q_{min} = 2678$ s: (a) – safe trajectory of own ship, (b) – changes in the rudder angle, (c) – changes in the rotational speed of the propeller.*
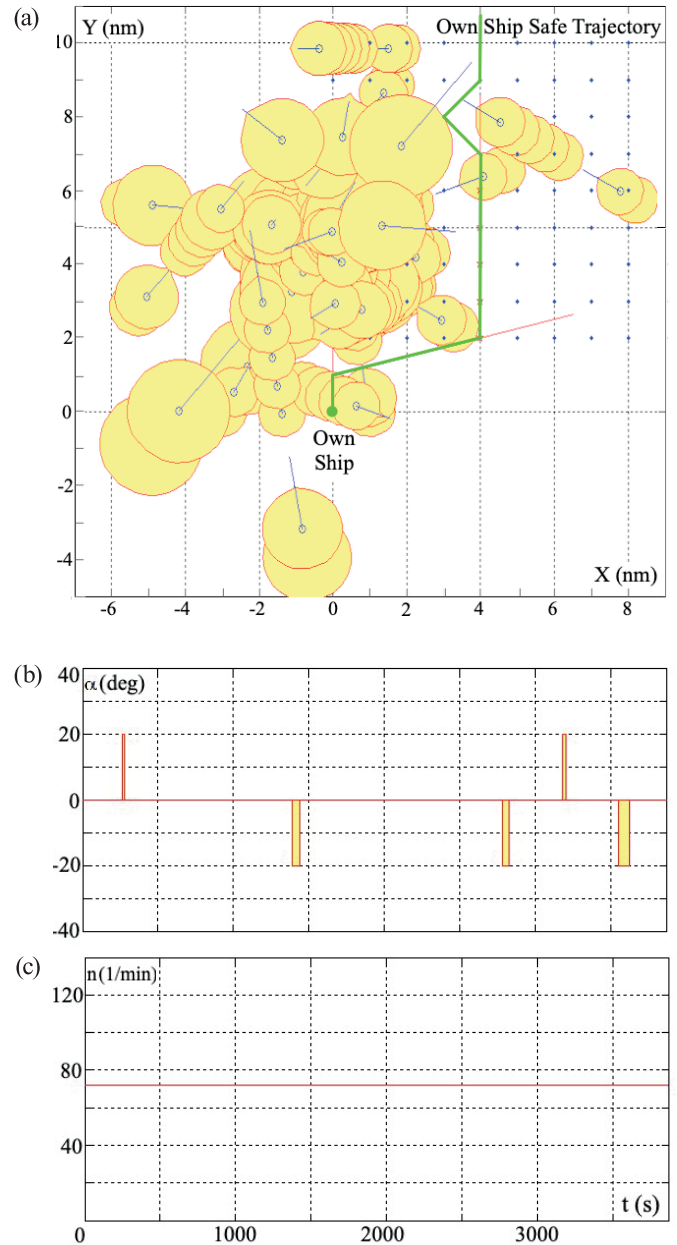

Figure 7. Computer simulation results for the neuro-optimal safety of a ship's trajectory and control sequence under conditions of restricted visibility at sea for $D_s = 1.5$ nm, with circular domains, $Q_{min} = 4033$ s: (a) – safe trajectory of own ship, (b) – changes in the rudder angle, (c) – changes in the rotational speed of the propeller.

COMPUTER SIMULATION OF A GAME-OPTIMAL SAFE TRAJECTORY THAT CONSIDERS THE UNCERTAINTY IN THE NAVIGATIONAL SITUATION

According to Lisowski [15], Song [23], Szlapczynska et al. [25] and Wang [26], the definition of the collision avoidance problem may be obvious, ignoring data sensitivity which may result from external elements dictated by climatic conditions and the state of the ocean, insufficient knowledge concerning other ships and imprecise proposals of international conflict of law rules (COLREGs).

The first approach, which involves using the theory of differential games in steering as described by Reddy et al. [21], allows for control of the ship to be considered under conditions of uncertainty in navigational situations.

Single procedure models have been helpful in training students as future seagoing officers and allowing for the selection of algorithms for safe ship control in collision situations. For the practical synthesis of control algorithms, the positional and matrix game models presented by Sanchez-Soriano [22] can be used.

The model that is closest to real situations is the matrix game model, which uses a risk matrix of ship collisions to identify various manoeuvring strategies in terms of changing courses or speeds.

The control variables of the ship are represented by the course ψ and the speed V, while for s met ship by course y_s and speed V_s. The state variables for the ship are represented by the risk of collision r_s, while for the met ship by distance D_s and bearing N_s.

The risk of collision with the s-th ship encountered, r_s, is a relative assessment of the current situation of the approach characterised by the quantities $D_{s,min}$ and $T_{s,min}$, compared to the expected safe situation characterised by the previously adopted safe values of D_{safe} and T_{safe}.

In most real control processes, the matrix game does not reach the saddle point, and does not guarantee balance when used in pure object strategies. The approximate solution of the real game, according to Osborne [20], constitutes a component of the mixed strategy, which expresses the probability distribution p_s of the players' pure strategies.

The optimal game control of the ship is the strategy of the highest probability to use is as follows:

$$u_{0,opt} = u_0 \left[\left(p_s^{(\sigma_0, \sigma_s)} \right)_{max} \right] \quad \textbf{(9)}$$

where:

σ_0 – single strategies for controlling the ship's movement to avoid collisions; in game theory, these are called 'pure strategies',

σ_s – single strategies for controlling the movement of the s-th ship that is cooperating to avoid collisions or scenarios leading to collisions for various reasons; in game theory, these are called 'pure strategies'.

In a non-cooperative game, the quality index Q for optimal ship control can be formulated as follows:

$$Q^{n-c}(\sigma_0, \sigma_s) = \min_{\sigma_0} \max_{\sigma_s} p_s(\sigma_0, \sigma_s) \quad \textbf{(10)}$$

However, in a cooperative game, this can be expressed as follows:

$$Q^c(\sigma_0, \sigma_s) = \min_{\sigma_0} \min_{\sigma_s} p_s(\sigma_0, \sigma_s) \quad \textbf{(11)}$$

Our computer simulation of cooperative and non-cooperative game control algorithms was implemented in Matlab for the navigational situation illustrated in Figure 5.

Figures 8 and 9 show the optimal safe voyage routes under good and restricted shipping conditions, respectively, for cooperation and no cooperation between ships in terms of avoiding collisions.
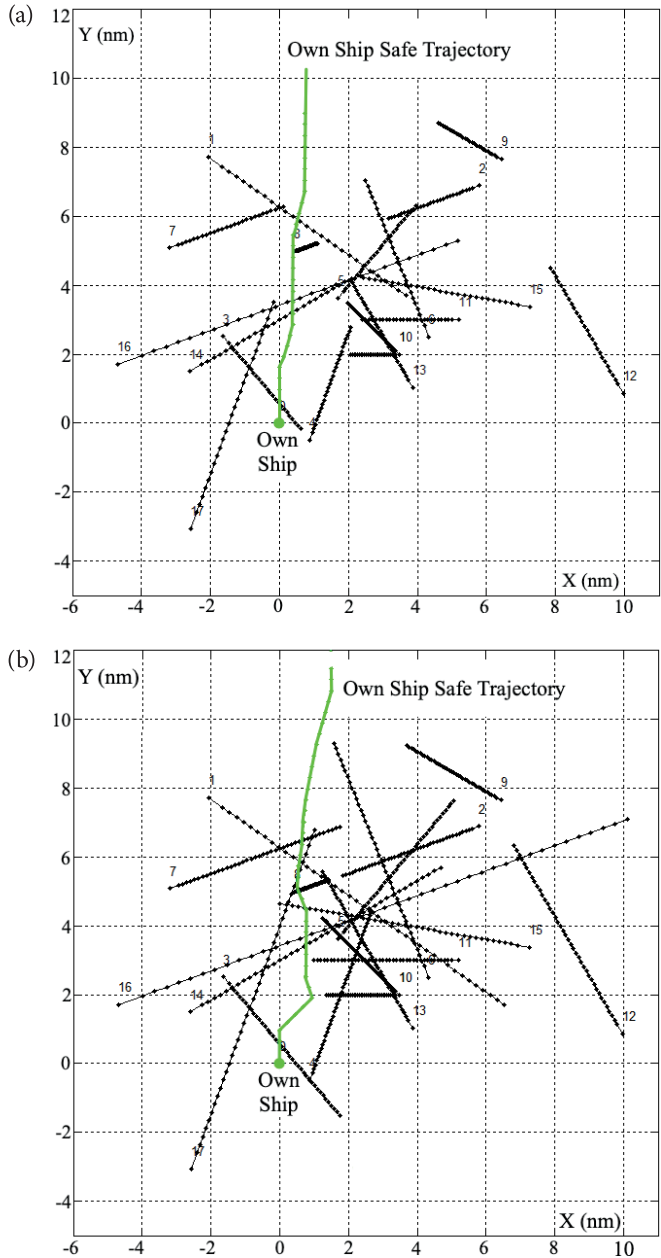

Fig. 8. Game-optimal safe ship trajectories in a computer simulation of navigational in good shipping conditions for Ds = 0.5 nm: (a) in a cooperative matrix game, Q_{min} = 3312 s; and (b) in a non-cooperative matrix game, Q_{min} = 3660 s.
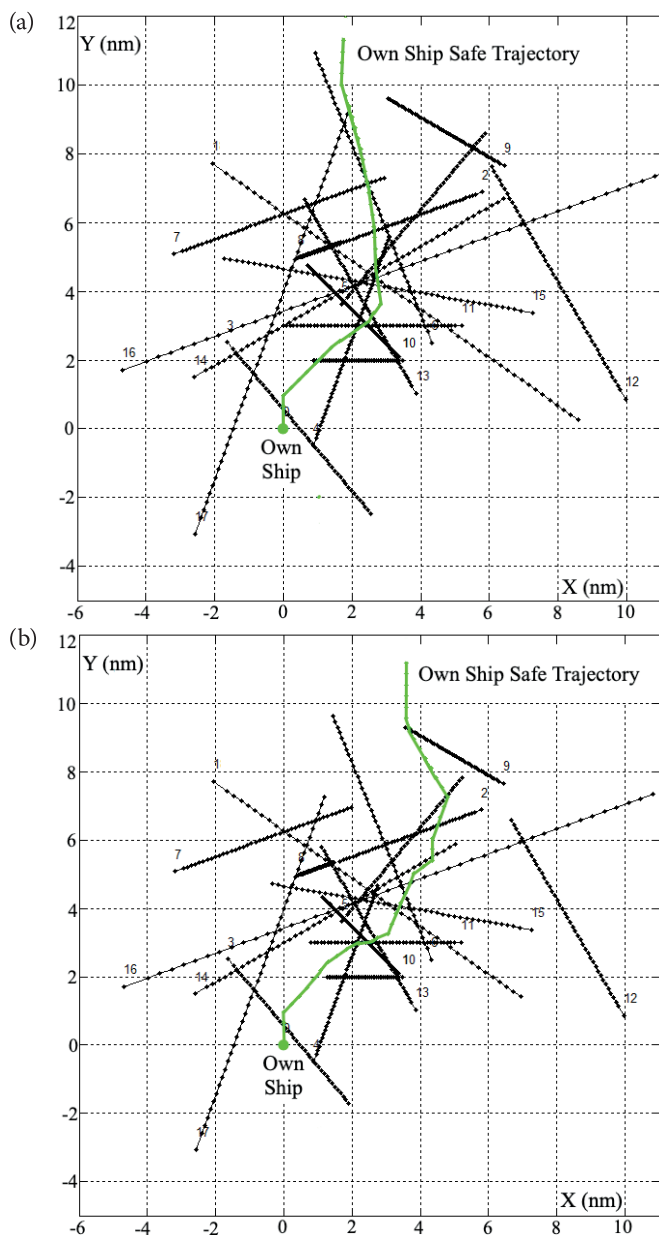
Fig. 9. Game-optimal safe ship trajectories in a computer simulation of navigation with restricted shipping conditions for D_s = 1.5 nm: (a) in a cooperative matrix game, Q_{min} = 3990 s; and (b) in a non-cooperative matrix game, Q_{min} = 4896 s.

CONCLUSIONS

An analysis of our research results in an effort to determine a safe and optimal route for a ship based on the use of selected computational intelligence methods allows us to draw the following conclusions.

In terms of calculating the neuro-optimal safe trajectory, the use of ARPA radar to identify object movement parameters allowed us to develop an algorithm to support the navigator in determining a safe trajectory, as a sequence of changes in the course and speed of the ship. Representing the movements of encountered ships in the form of moving

neural domains of variable size, depending on the distance and time between approaching ships, allowed us to take into account the subjectivity of the navigator when assessing the risk of collision. The use of several hundred navigator officers to teach an artificial neural network allowed the computational algorithm to interpret the domain in which there is a danger of encountering ships better than if only a single experienced navigator had been used. An analysis of the possible domain shapes shows that they can be adapted to open or restricted waters. The node density in the dynamic programming trajectory of the ship is a compromise between calculation time and ship-route accuracy.

When calculating the game-optimal safe trajectory, our algorithm takes into account both the COLREG rules when starting the game and the dynamics of the ships, in the form of the advance manoeuvring time, their degree of cooperation and the end game when the risk of collision becomes zero. A new definition of ship collision risk was also presented here based on two assessments of the same navigational situation: the real situation regarding the proximity of objects, and the safe situation as determined by the reference parameters.

This work does not cover all of the issues associated with the safe management of the movement of ships at sea. Subsequent studies will include an analysis of the sensitivity of safe ship control to inaccurate information from navigation devices, changes in the parameters of the ship dynamics, and the impact of hydrometeorological disturbances.

Future papers on computational intelligence in marine control engineering education should focus on additional groups of ship officers, such as mechanics and electricians.

ACKNOWLEDGEMENTS

REFERENCES

1. J.H. Ahn, K.P. Rhee, and Y.J You, "A study on the collision avoidance of a ship using neural networks and fuzzy logic," *Applied Ocean Research,* vol. 37, pp. 162–173, 2012. DOI: 10.1016/j.apor.2012.05.008

2. R.E. Bellman, *Dynamic Programming.* New York: Dover Publications, 2003. ISBN 0-486-42809-5

3. M. Borrego, E.P. Douglas, and C.T. Amelink, *Quantitative*, "Qualitative and mixed research methods in engineering education," *Journal of Engineering Education*, vol. 98, no. 1, pp. 53–66, 2009. DOI: 10.1002/j.2168-9830.2009.tb01005.x

4. R. Cwilewicz and J. Lisowski, "The integrated maritime education and research activity of Gdynia Maritime

University," in *12th Annual General Assembly of IAMU - Green Ships, Eco Shipping, Clean Seas*, Gdynia Maritime University, Gdynia, 17 June 2011, pp. 87–98.

5. B. Guenin, J. Konemann, and L.A. Tuncel, *Gentle Introduction to Optimization*. Cambridge, UK: Cambridge University Press, 2014. ISBN 978-1-107-05344-1

6. S.S. Guzey and M. Aranda, "Student participation in engineering practices and discourse: An exploratory case study," *Journal of Engineering Education*, vol. 106, no. 4, pp. 585–606, 2017. DOI: 10.1002/jee.20176

7. H. Heiselberg and A. Stateczny, "Remote sensing in vessel detection and navigation," *Sensors*, vol. 20, pp. 1–9, 2020. DOI: 10.3390/s20205841

8. M. Henri, M.D. Johnson, and B. Nepal, "A review of competency-based learning: Tools, assessments, and recommendations," *Journal of Engineering Education*, vol. 106, no. 4, pp. 607–638, 2017. DOI: 10.1002/jeee.20180

9. L. Hongguang and Y. Yong, "COLREGS-constrained real-time path planning for autonomous ships using modified artificial potential fields," *Journal of Navigation*, vol. 71, pp. 1–21, 2018. DOI: 10.1017/S0373463318000796

10. Y. Huang, L. Chen, P. Chen, R.R. Negenborn, and P.H.A.J.M. van Gelder, "Ship collision avoidance methods: State-of-the-art," *Safety Science*, vol. 121, pp. 451–473, 2020. DOI: 10.1016/j.ssci.2019.09.018

11. K.S. Kula, "Automatic control of ship motion conducting search in open waters," *Polish Maritime Research*, vol. 27, no. 4, pp. 157-169, 2020. DOI: 10.2478/pomr-2020-0076

12. L.R. Lattuca, D.B. Knight, H.K. Ro, and B.J. Novoselich, "Supporting the development of engineers' interdisciplinary competence," *Journal of Engineering Education*, vol. 106, no. 1, pp. 71–97, 2017. DOI: 10.1002/jeee.20155

13. 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. DOI: 10.1109/ACCESS.2020.3043539

14. A. Lebkowski, "Evolutionary methods in the management of vessel traffic," in *Proc. Int. Conf. on Marine Navigation and Safety of Sea Transportation*, Gdynia, Poland, 17 June 2015, pp. 259–266. DOI: 10.12716/1001.12.01.13

15. J. Lisowski, "Multi-criteria optimization of multi-stage positional game of vessels," *Polish Maritime Research*, vol. 27, no. 1, pp. 46-52, 2020. DOI: 10.2478/pomr-2020-0005

16. Z. Liu, Z. Wu, and Z. Zheng, "A cooperative game approach for assessing the collision risk in multi-vessel encountering," *Ocean Engineering*, vol. 187, pp. 1–12, 2019. DOI: 10.1016/j.oceaneng.2019.106175

17. Z. Liu, "Pre-filtered backstepping control for underactuated ship path following," *Polish Maritime Research*, vol. 26, no. 2, pp. 68-75, 2019. DOI: 10.2478/pomr-2019-0026

18. S. Nikolic, "Improving the laboratory learning experience: A process to train and manage teaching assistants," *IEEE Transaction on Education*, vol. 58, no. 2, pp.130–139, 2015. DOI: 10.1109/TE.2014.2335712

19. N.S. Nise, *Control Systems Engineering*. New York: John Wiley & Sons, 2019. ISBN 978-1-119-72140-6

20. M.J. Osborne, *An Introduction to Game Theory*. New York: Oxford University Press, 2004.

21. P.V. Reddy and G. Zaccour, "Feedback Nash equilibria in linear-quadratic difference games with constraints," *IEEE Transactions on Automatic Control*, vol. 62, pp. 590–604, 2016. DOI: 10.1109/TAC.2016.2555879

22. J. Sanchez-Soriano, "An overview of game theory applications to engineering," *International Game Theory Review*, vol. 15, pp. 1–18, 2013. DOI: 10.1142/S0219198913400197

23. L. Song, H. Chen, W. Xiong, et al., "Method of emergency collision avoidance for unmanned surface vehicle (USV) based on motion ability database," *Polish Maritime Research*, vol. 26, no. 2, pp. 55-67, 2019. DOI: 10.2478/pomr-2019-0025

24. J.L. Speyer and D.H. Jacobson, *Primer on Optimal Control Theory*. Toronto, Canada: SIAM, 2010. ISBN 978-0-898716-94-8

25. J. Szlapczynska and R. Szlapczynski, "Preference-based evolutionary multi-objective optimization in ship weather routing," *Applied Soft Computing*, vol. 84, pp. 1–21, 2019. DOI: 10.1016/j.asoc.2019.105742

26. S. Wang, Y. Tuo, "Robust trajectory tracking control of underactuated surface vehicles with prescribed performance," *Polish Maritime Research*, vol. 27, no. 4, pp. 148-156, 2020. DOI: 10.2478/pomr-2020-0075

27. J. Trevelyan, "Technical coordination in engineering practice," *Journal of Engineering Education*, vol. 96, no. 3, pp. 191–204, 2007. DOI: 10.1002/j.2168-9830.2007.tb00929.x

28. H.J. Trussell and E.J. Dietz, "A study of the effect of graded homework in a preparatory math course for electrical engineers," *Journal of Engineering Education*, vol. 92, no. 2, pp. 141–146, 2003. DOI: 10.1002/j.2168-9830.2003.tb00752.x

29. T.F. Weisner and W. Lan, "Comparison of student learning in physical and simulated unit operations experiments," *Journal of Engineering Education*, vol. 3, no. 3, pp. 5–12, 2004. DOI: 10.1002/2168-9830.2004.tb00806.x

30. A. Witkowska and R. Smierzchalski, "Adaptive dynamic control allocation for dynamic positioning of marine vessel based on backstepping method and sequential quadratic programming," *Ocean Engineering*, vol. 163, pp. 570–582, 2018. DOI: 10.1016/j.oceaneng.2018.05.061

31. J. Yong, *Optimization Theory – A Concise Introduction*. New Jersey: World Scientific, 2018. ISBN 978-981-3237-64-3

32. J. Zhuang, L. Zhang, Z. Qin, et al., "Motion control and collision avoidance algorithms for unmanned surface vehicle swarm in practical maritime environment," *Polish Maritime Research*, vol. 26, no. 1, pp. 107-116, 2019. DOI: 10.2478/pomr-2019-0012

CONTACT WITH THE AUTHOR

Józef Lisowski
e-mail: j.lisowski@we.umg.edu.pl

Uniwersytet Morski w Gdyni,
Morska 83,
81-225 Gdynia,
Poland