

Article

Optimality of Safe Game and Non-Game Control of Marine Objects

Józef Lisowski 

Faculty of Electrical Engineering, Gdynia Maritime University, 81-225 Gdynia, Poland; j.lisowski@we.umg.edu.pl; Tel.: +48-694-458-333

Abstract: The article presents a model of the process of safe and optimal control of an autonomous surface object in a group of encountered objects. An algorithm for determining the optimal and safe trajectory based on a multi-object game model was proposed, and an algorithm for determining the optimal trajectory was proposed for comparative analysis, not taking into account the maneuverability of other objects. Simulation studies of the algorithms made it possible to assess the optimality of the trajectories for various acceptable object strategies. An analysis of the characteristics of the sensitivity of the safe control—assessed with the risk of collision, both on the inaccuracy of navigation data and on the number of possible strategies of objects, was carried out.

Keywords: optimal control; game control; sensitivity control; computer simulation

1. Introduction

For the current study, the abbreviations and acronyms are summarized in Abbreviations.

1.1. State of Knowledge

The literature review was divided into the following thematic groups:

- description of basic ASV functions [1–3];
- use of the ASV to detect, recognize and track other objects [4,5];
- ASV control methods [6–8];
- ASV route planning [9–15];
- directing the ASV traffic taking into account COLREGs [16–20].

One of the oldest reviews of the state of development of autonomous surface vehicles control is the work of Zhao et al. [1] describing the autonomous surface units (ASCs), also called autonomous surface vehicles (ASVs), as a type of autonomous marine vehicle without direct human control.

The transition from a research concept to a commercial product and related services of unmanned surface vehicles (USVs) in shipping was described by Barrera et al. [2], showing a multidisciplinary approach to this field.

Choi et al. [3] presented a test stand for the validation of basic navigation technologies for autonomous marine robots in the field of control by tracking waypoints and avoiding obstacles. Two methods of underwater location were used, in the form of acoustic navigation based on the Kalman filter and navigation based on geophysics using a particle filter.

Much work has been devoted in recent years to the use of autonomous surface vehicles for the detection, recognition and tracking of various objects. Thus, Chen et al. [4] presented a solution for an autonomous USV that has the ability to acquire various types of data and information in sea areas, process them and then implement the mission, using a deep convolutional neural network to identify approaching vehicles in order to alert the ground station or study the surrounding environment-assigned locations.

The design of a surface vehicle capable of detecting objects at the bottom of a larger reservoir, navigating in their direction and picking them up using the attached grapple was



Citation: Lisowski, J. Optimality of Safe Game and Non-Game Control of Marine Objects. *Electronics* **2023**, *12*, 3637. <https://doi.org/10.3390/electronics12173637>

Academic Editors: Mahmut Reyhanoglu, Erkan Kayacan and Mohammad Jafari

Received: 12 July 2023

Revised: 21 August 2023

Accepted: 27 August 2023

Published: 28 August 2023



Copyright: © 2023 by the author. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

presented by Sneha [5]. Control is done first with a PID controller for proof of concept, then with an LQR controller and observer for optimal control.

Omrani et al. [6] presented the use of an ASV aircraft for monitoring marine facilities and ports by implementing a stereovision system for detecting and tracking static and dynamic obstacles.

Zhang et al. [7] proposed a method of accurate target detection with long-strip targets on the water, based on a convolutional neural network, for detecting and tracking targets in the processes of sea exploration and protection. Using a dicyclic loop PID control scheme, the autonomous surface vehicle is steered to approach a long-strip target with a near-optimal path design.

In addition, Lee and Lin [8] designed a deep learning-based surveillance and reconnaissance system for unmanned ground vehicles by adopting a Siamese network as the main neural network architecture to achieve target tracking.

Currently, there is a growing interest in autonomous ships that reduce human error and support intelligent navigation, where automatic collision avoidance and route planning are key issues, especially in restricted waters. To solve this problem, Hongguang and Yong [9] proposed a path-guided hybrid artificial potential field method that allows the rapid generation of adaptive, collision-free and COLREGs rules-limited trajectories in restricted waters using deterministic calculations.

Zhou et al. [10] reviewed research on the route planning of USVs based on multimodality constraint, which can be divided into route planning, trajectory planning and traffic planning. However, Due et al. [11] analyzed the progress of route planning research based on the multimodality constraint, which can be divided into the following stages: route planning, trajectory planning and traffic planning.

Martins et al. [12] designed a docking system for an ASV with an AUV in a river environment.

Park et al. [13] described object recognition based on images from several cameras in order to detect obstacles on autonomous ships, and then track the movements of recognized ships.

However, Li et al. [14] designed a USV and a UAV path-following control system in the presence of structural uncertainties and external disturbances, consisting of three-dimensional mapping guidance and an adaptive fuzzy control algorithm.

Wang et al. [15] introduce Roboat's autonomy system for urban waterways, based on the extended Kalman filter, calculation of optimal trajectories to avoid static and dynamic obstacles, and predictive steering to accurately track the trajectory from the planner in rough water.

Hongguang and Yong [16] presented a deterministic control method of real-time route planning for autonomous ships or USVs, taking into account the function of the repulsive potential field and the corresponding virtual forces, constrained by COLREGs for own ship actions.

Recently, unmanned USVs have been actively developed, ensuring traffic safety by preventing collisions with other vessels by taking into account COLREGs. Thus, Kim et al. [17] proposed an algorithm that predicts dangerous situations based on the DCPA and the TCPA.

However, Zhong et al. [18] proposed an ontological model of ship behavior based on COLREGs using knowledge graph techniques, which aims to help the machine interpret the COLREGs; the ship is perceived as a spatiotemporal object, and its behavior is described as changing the elements of the object on spatiotemporal scales using a resource description framework and function and method mappings for set expressions.

Moreover, Hu et al. [19] recently reviewed recent advances in COLREGs-compliant ASV navigation from a traditional approach to a learning-based approach in implementing the three steps of safe navigation, namely from collision detection to decision control making and then rerouting.

Moreover, the topic of ensuring the safety of the movement of a USV moving in a group of other USVs has been raised.

Sun et al. [20] proposed a self-organizing cooperative control strategy for multiple USVs to pursue intelligent escapes in the event of a dynamic obstacle craft. In order to better deal with dynamic obstacle ships encountered in the game pursuit process in a dynamic ocean environment, the collision avoidance method is based on the artificial field potential for ships between USVs and the strategy for dynamic obstacle ship collision avoidance is based on COLREGs.

1.2. Paper Thesis and Objectives

The analysis of the literature review shows that so far, the problem of the dependence of the level of game control motion security of the autonomous surface object acting in the group of other encountered objects, both on the inaccuracy of navigation information and on the range of acceptable control strategies, has not been addressed.

Therefore, the thesis of the article is to show that by analyzing the sensitivity of the collision risk, it is possible to assess the range of acceptable values, both the inaccuracy of individual components of navigation information and the number of acceptable control strategies.

The scientific goal is to analyze the game and optimal control sensitivity to changes in the state and control process of an autonomous surface object movement in a group of other encountered objects. The measure of the game control is the collision risk value, and the measure of optimal control is the final deviation of the trajectory from its predetermined direction.

The aim of the research is an experimental comparative analysis of the game control with the non-game control with a different number of acceptable strategies for controlling objects.

A significant contribution of the author of the article in solving the research task consists in: synthesis of a computer program for calculating the safe path of an autonomous surface object through a group of other autonomous surface objects in game environment conditions; assessing the impact of the number of allowed strategies on the control optimization result; formulation of data measurement accuracy requirements based on the sensitivity analysis of the game control.

The progress of this work is due to the fact that it is one of the game control layers of the integrated multi-layer surface objects control system. This system consists of the following layers: optimization path by evolutionary algorithm; safe control by dynamic programming or ant colony or fuzzy neural algorithms; robust control by linear matrix inequalities or predictive line of sight algorithms; and the direct control by internal model control or PID adaptive autopilot [21].

1.3. Paper Content Plan

The paper is organized as follows. First, the autonomous control process of a surface object is described in the form of state equations, constraints and control objective functions. The next section describes the author's safe trajectory algorithms, first the game control algorithm and then, for comparison, the non-game control algorithm. The next section presents the results of a computer simulation of the developed algorithms on the example of a real navigational situation at sea. The test results as a comparison of trajectories optimality and safe control sensitivity characteristics are illustrated. The conclusion summarizes the results of the research and presents the scope for future work on the subject of this paper.

2. Autonomous Surface Object Control Process

The control process in a group of autonomous surface objects, where our autonomous surface object 0 controlled avoiding collisions by the change in course ψ_0 as control u_0 is in position (X_0, Y_0) and other k autonomous surface objects controlled by their course ψ_k as control u_k are in position (X_k, Y_k) , is presented in Figure 1.

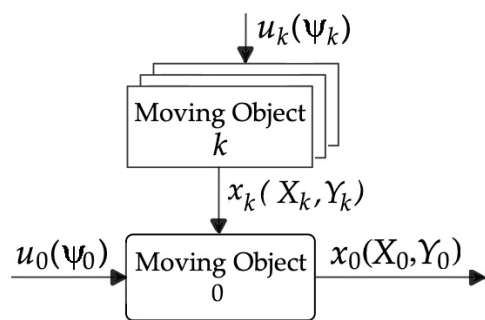


Figure 1. Graphical model of a group game involving autonomous surface objects.

The controls of individual objects affect their relative movement and the distance at which they will pass each other, which becomes the basis for the model involving a cooperative game or a non-cooperative game. When following the COLREGs, we are dealing with a cooperative game.

However, difficult environmental conditions, disturbances in measurements of data-controlled autonomous objects, and various subjective factors make up a non-cooperative game.

The control process of our autonomous surface object 0 in the k autonomous surface objects group, shown in Figure 2, consists of the state equations of this process:

$$\dot{x}(t) = f(x,u,t) \tag{1}$$

where $x(D_k, N_k, X_k, Y_k)$ are state variables, $u(\psi_k)$ is a control variable, $k = 1, 2, \dots, K$ —number of autonomous objects, D_k and N_k are the distance from and bearing to the k autonomous surface object, X_k and Y_k are position coordinates of k autonomous surface objects and ψ_k is the course of the k autonomous surface object.

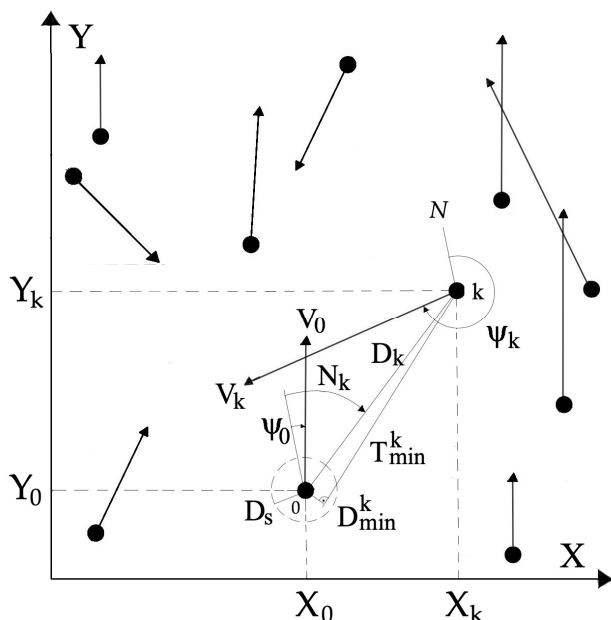


Figure 2. Variables describing the movement of a group of autonomous surface objects: V_0 and ψ_0 are, respectively, the speed and the course of our autonomous surface object; V_k and ψ_k are, respectively, the speed and the course of another k autonomous surface object; D_k and N_k are, respectively, the distance from and the bearing to the k autonomous surface object; D_{min}^k and T_{min}^k stand, respectively, for the minimum distance from and the time of passing autonomous surface objects; D_s is the safe passing distance of autonomous surface objects; and (X, Y) are the position coordinates of autonomous surface objects.

State and control constraints in general notation have the form:

$$g(x, u, t) \geq 0 \tag{2}$$

while here they result from the need to keep a safe distance between objects $D_{min}^k \geq D_s$:

$$D_{min}^k(V_0, V_k, \psi_0, \psi_k, X_k, Y_k) - D_s \geq 0 \tag{3}$$

where D_{min}^k is the shortest passing distance of our and the k autonomous surface object, D_s is the safe distance of passing objects, depending on the state of visibility at sea, ψ_0 —the course of our autonomous surface object.

Control objective function as an index of optimal control quality in general form:

$$Q = Q(x, u, t) \rightarrow \min \tag{4}$$

takes the form of r_k collision risk in the process of safe objects control:

$$Q_0^k = r_0^k(D_{min}^k, D_s, T_{min}^k, T_s, D_k) \rightarrow \min \tag{5}$$

where T_{min}^k is the shortest passing time of our and the k autonomous surface object, T_s is the safe time of passing objects, depending on the state of visibility at sea.

At present, the influence of sea dynamics on the controller’s control process is taken into account in accordance with the applicable COLREGs, by selecting an adequate value of the safe distance between objects to the state of sea waves. With sea states in the range of 0 to 5° B corresponding to good visibility at sea, the value of the safe passing distance D_s should be set in the range of 0.1 to 1 nm. However, for rough seas from 6 to 12° B, corresponding to restricted visibility at sea, a safe passing distance D_s of 1 to 3 nm should be assumed. In future works, the components of the disturbance vector from waves, wind and the sea current should be taken into account directly in the mathematical model of the process [22,23].

3. Algorithms for Determining a Safe Trajectory

3.1. Game Control Algorithm

Our autonomous surface object has control $u_0(\Delta\psi_0^i)$, where $i = 1, 2, \dots, I$ is the number of strategies it is allowed to change course to carry out an anti-collision maneuver at a distance not less than D_s (see Figure 3). Similarly, at each k , another autonomous surface object uses control $u_k(\Delta\psi_k^j)$, where $j = 1, 2, \dots, J$ is the number of strategies it is allowed to change course to carry out an anti-collision maneuver.

A collision risk matrix $R[r_k(\Delta\psi_0^i, \Delta\psi_k^j)]$ is created, where the risk of collision r_k is the relative measure of safety when autonomous objects pass each other. The current situation CS is described by the values $CS(D_{min}^k, T_{min}^k, D_k)$. The safe situation SS is defined by the quantities $SS(D_s, T_s)$. The author of this paper in [24] defines the collision risk r_k as the mean squared reference measure of the assessment of the current situation CS of the proximity of objects to the assessment of the expected safe situation SS:

$$r_k = \left[c_d \left(\frac{D_{min}^k}{D_s} \right)^2 + c_t \left(\frac{T_{min}^k}{T_s} \right)^2 + c \left(\frac{D_k}{D_s} \right)^2 \right]^{-\frac{1}{2}} \tag{6}$$

where c_d , c_t , and c —weighting factors depend on environmental conditions, with values from 0 to 1; for example, in situations of concentrated objects movement: $c_d = c_t = 0.4$, $c = 0.5$ and in situations of greater distances between objects: $c_d = c_t = 0.5$, $c = 0.1$.

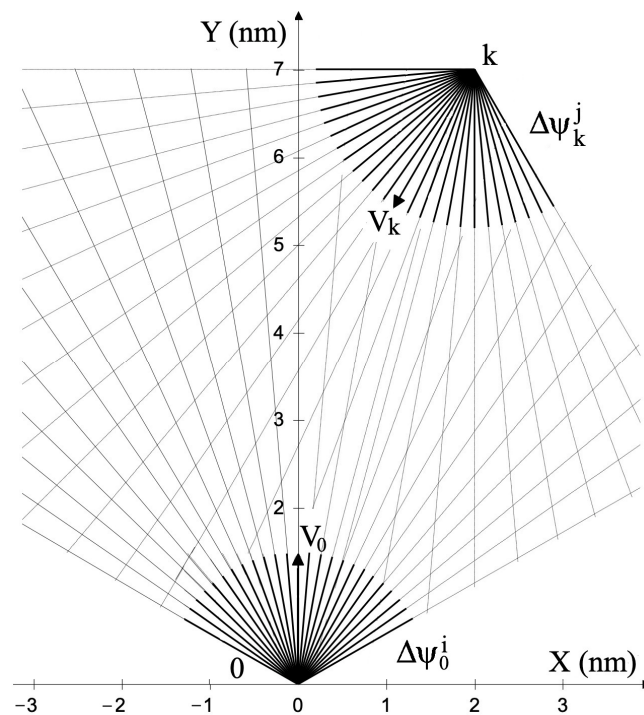


Figure 3. Sets of the number of acceptable strategies i for our 0 autonomous surface object and number of acceptable strategies j for another k autonomous surface object.

The safe and optimal game trajectory for our autonomous surface 0 object in a group of k other autonomous surface objects can be computed by dual linear programming:

$$Q_{0,k}^G = \min_i \max_j r_k \tag{7}$$

It is assumed that for various unknown reasons, other k autonomous objects lead to a collision using strategy j , maximizing the risk of collision. Its own autonomous surface object, for such a strategy of another object, then determines its area of acceptable strategies and from there selects the strategy i minimizing the risk of collision and implementing the anti-collision maneuver.

Another possibility to solve this control task is to apply the Bellman optimality principle, presented in the works of the author [25,26]. Regardless of the state and initial decisions of moving object 0, the remaining optimal strategies depend on the current state and the decision made. The state constraints of the control process consist of collision hazard domains by passing k moving objects, in the shapes of a circle and hexagon, changing in size depending on the risk of collision. The domains are generated by a three-layer artificial neural network previously learned by the navigators.

The operation of the game control algorithm G developed by the author of this article for determining the safe trajectory of an autonomous surface object in a group of autonomous surface objects as a multi-stage decision-making process was tested in MATLAB/Simulink version R2023a software (Algorithm 1).

The original, innovative algorithm G part is the formation of a collision risk matrix based on the current information about the navigation situation from the ARPA anti-collision system, and then applying dual linear programming to it to find optimal strategies in each subsequent step of the game flow. In this way, a multi-step game control algorithm is synthesized. Therefore, Algorithm G computes the elements of the collision risk matrix, where the number of rows equals the number of our autonomous surface object strategies and the number of columns equals the total number of strategies of all other autonomous surface objects.

Algorithm 1: Game control of autonomous surface object

BEGIN**Input** and development of initial data**Display** of the navigation situation from: GPS, ARPA, Log, Gyro**Calculation** of collision risk matrix $R(r^i_0, r^j_k)$;**Determination** of most dangerous another k autonomous surface object;**Calculation** of safe course of our 0 autonomous surface object by dual linear programming;**Calculation** of dangerous courses of another k autonomous surface object dual linear programming;**Designation** of total safe course of our 0 autonomous surface object in relation to all K encountered objects;**IF** object: $k \neq K$ **THEN****GOTO** Determination of most dangerous other k autonomous surface object**ELSE IF** Stage: $s \neq S$ **THEN****GOTO** Calculation of collision risk matrix $R(r^i_0, r^j_k)$;**ELSE** Trajectory visualization of autonomous surface objects group;**END**

3.2. Non-Game Control Algorithm

Assuming that other autonomous objects move without changing course over time, the game control task is reduced to the non-game control task:

$$Q_{0,k}^{NG} = \min_i r_k \quad (8)$$

The NG algorithm is used in this paper only for the comparative analysis of the game control with the classic optimal control. The operation of the non-game control algorithm of determining a safe trajectory in a group of autonomous surface objects as a multi-stage decision-making process was tested in MATLAB/Simulink version R2023a software (Algorithm 2).

Algorithm 2: Non-Game control of autonomous surface object

BEGIN**Input** and development of initial data from: GPS, ARPA, Log, Gyro;**Set designation** of permissible course maneuvers of our 0 autonomous surface object in relation to another k autonomous surface object;**Determination** of safe course of our 0 autonomous object from set of permissible maneuvers using linear programming;**IF** object: $k \neq K$ **THEN****GOTO**(Set designation of permissible course maneuvers of our 0 autonomous surface object in relation to another k autonomous surface object);**ELSE IF** Stage: $s \neq S$ **THEN****GOTO**(Set designation of permissible course maneuvers of our 0 autonomous surface object in relation to another k autonomous surface object);**ELSE** Trajectory visualization of autonomous surface objects group;**END**

4. Computer Simulation

4.1. Comparison Trajectories Optimality

The G and NG algorithms for calculating the safe path of our autonomous surface vehicle through a group of $k = 14$ autonomous surface targets were subjected to simulation tests on the example described in Table 1 and shown in Figures 4–8.

Table 1. Quantities describing the movement of a group of autonomous surface objects.

Autonomous Surface Object <i>k</i>	Speed V_k (kn)	Course ψ_k (deg)	Distance D_k (nm)	Bearing N_k (deg)
0	12	0	0	0
1	9	206	11.8	15
2	18	256	6.0	37
3	12	180	7.8	330
4	0	0	4.1	14
5	6	33	6.1	359
6	0	0	4.9	270
7	8	359	5.0	85
8	18	334	8.3	55
9	15	0	6.4	72
10	13	3	6.7	350
11	0	0	7.5	29
12	12	0	8.3	34
13	6	0	9.7	330
14	5	2	8.7	6

We assume that autonomous surface objects are equipped with regulators programmed in a microcontroller or programmable logic controller (PLC) and their motion control takes predetermined discrete values, which are called control strategies.

For the simulation tests, five sets of object-maneuvering strategies were adopted and are presented in Table 2.

Table 2. The number of strategies controlling the movement of the autonomous surface vehicles adopted for simulation tests.

Strategies Sets	Our 0 Autonomous Surface Object	Other <i>k</i> Autonomous Surface Object
A	$i = 2$ $u_0 = \Delta\psi_0^i = 0, 60^\circ$	$j = 3$ $u_k = \Delta\psi_k^j = -30^\circ, 0, 30^\circ$
B	$i = 3$ $u_0 = \Delta\psi_0^i = 0, 30^\circ, 60^\circ$	$j = 3$ $u_k = \Delta\psi_k^j = -30^\circ, 0, 30^\circ$
C	$i = 4$ $u_0 = \Delta\psi_0^i = 0, 20^\circ, 40^\circ, 60^\circ$	$j = 3$ $u_k = \Delta\psi_k^j = -30^\circ, 0, 30^\circ$
D	$i = 13$ $u_0 = \Delta\psi_0^i = 0 \div 60^\circ$ for each 5°	$j = 3$ $u_k = \Delta\psi_k^j = -30^\circ, 0, 30^\circ$
E	$i = 13$ $u_0 = \Delta\psi_0^i = 0 \div 60^\circ$ for each 5°	$j = 25$ $u_k = \Delta\psi_k^j = -60 \div 60^\circ$ for each 5°

For the validation of the results of the experiment evaluating the quality of the game control *G*, the most reliable and rational method of comparing them with the non-game control *NG* was adopted.

Comparing the results of experimental studies of the safe path algorithms of our autonomous surface object through a group of other autonomous surface targets, we can conclude the following:

- The greater the number of admissible strategies that was available for the objects, i.e., the greater the angular resolution of the course change, the smaller was the deviation *d* of the safe trajectory; for the non-game algorithm *NG* approximately three times, and for the game algorithm *G* approximately twice;
- For a small number of acceptable strategies, the deviation of the safe path is 30–60% greater for the *G* algorithm than for the *NG* algorithm;

- For more acceptable strategies, the safe path deviation becomes 200 ÷ 300% greater for the G algorithm than for the NG algorithm.

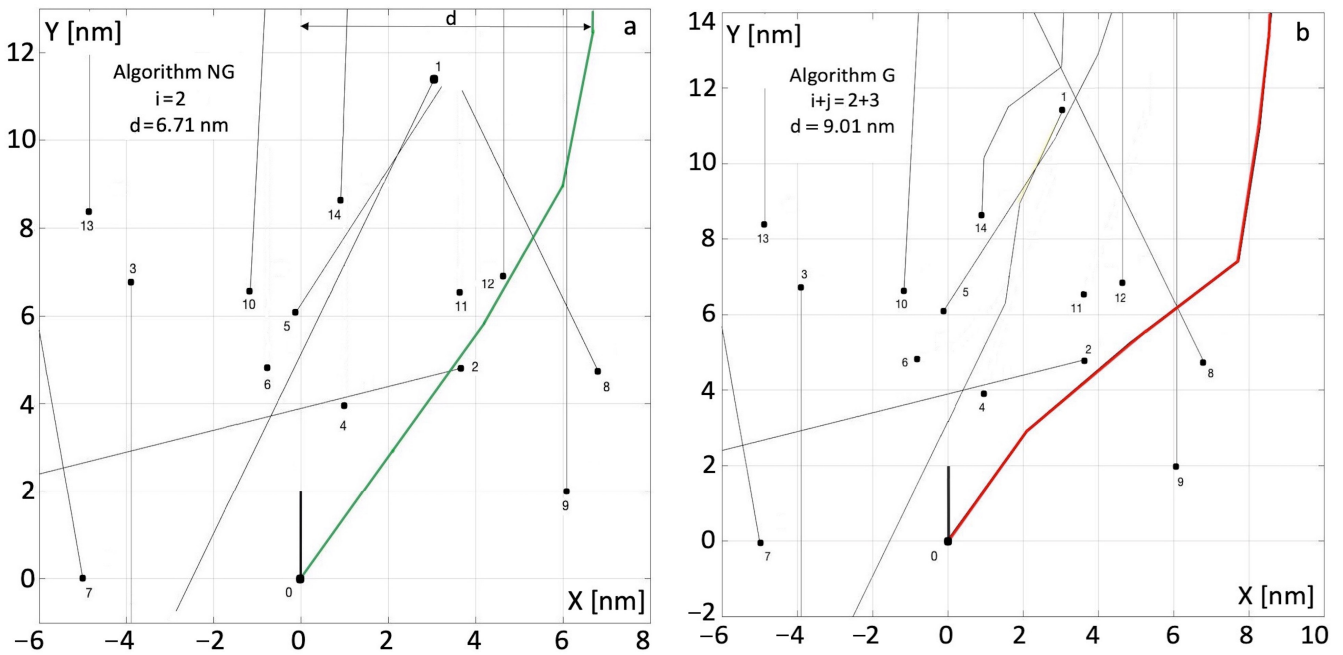


Figure 4. Safe trajectory of our autonomous surface object using set A of objects maneuvering strategies: (a) non-game control (green); (b) game control (red).

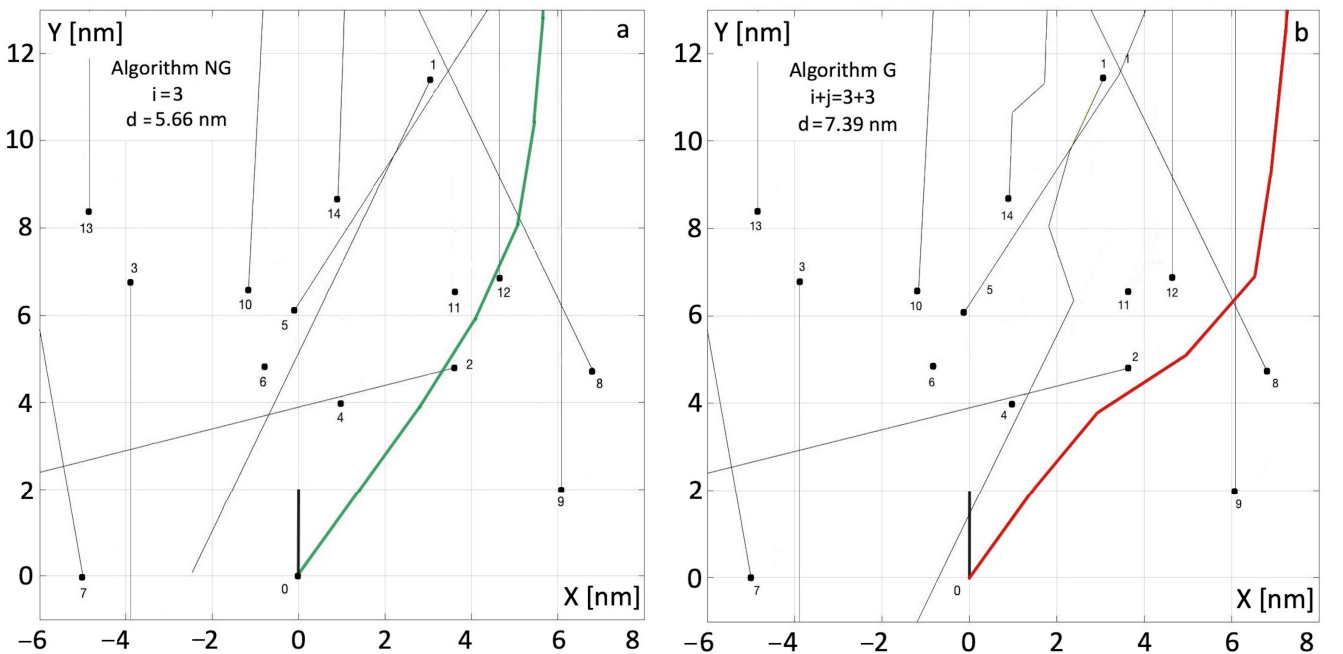


Figure 5. Safe trajectory of our autonomous surface object using set B of objects maneuvering strategies: (a) non-game control (green); (b) game control (red).

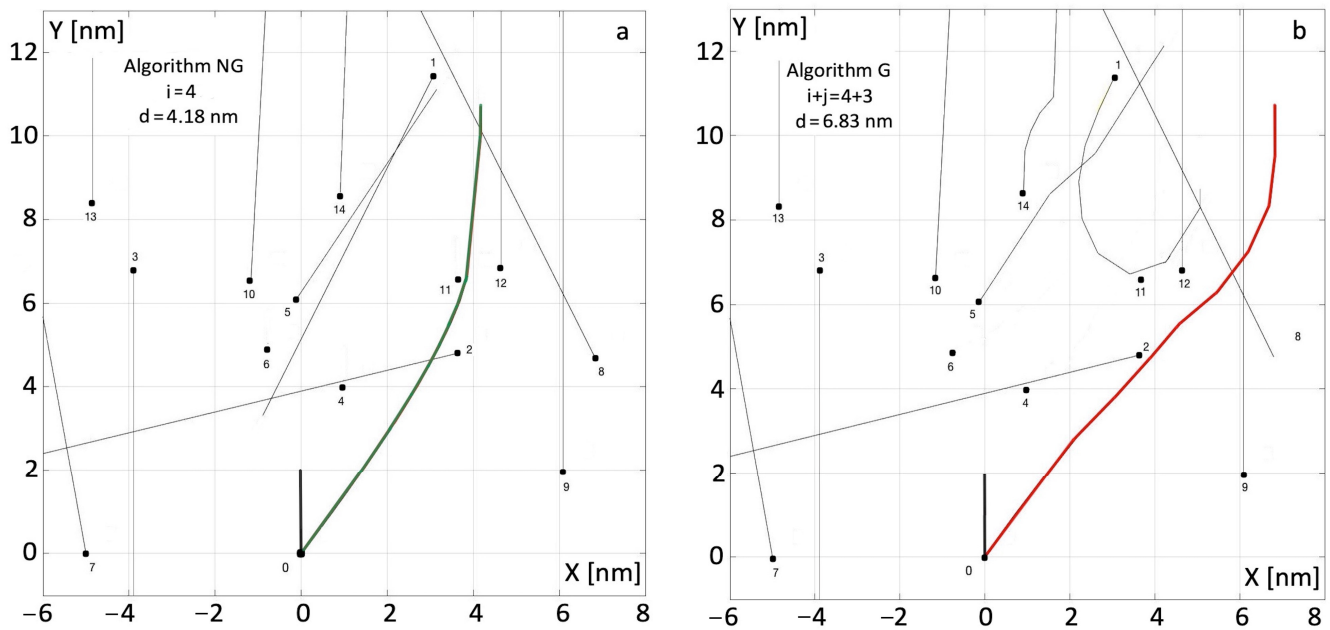


Figure 6. Safe trajectory of our autonomous surface object using set C of objects maneuvering strategies: (a) non-game control (green); (b) game control (red).

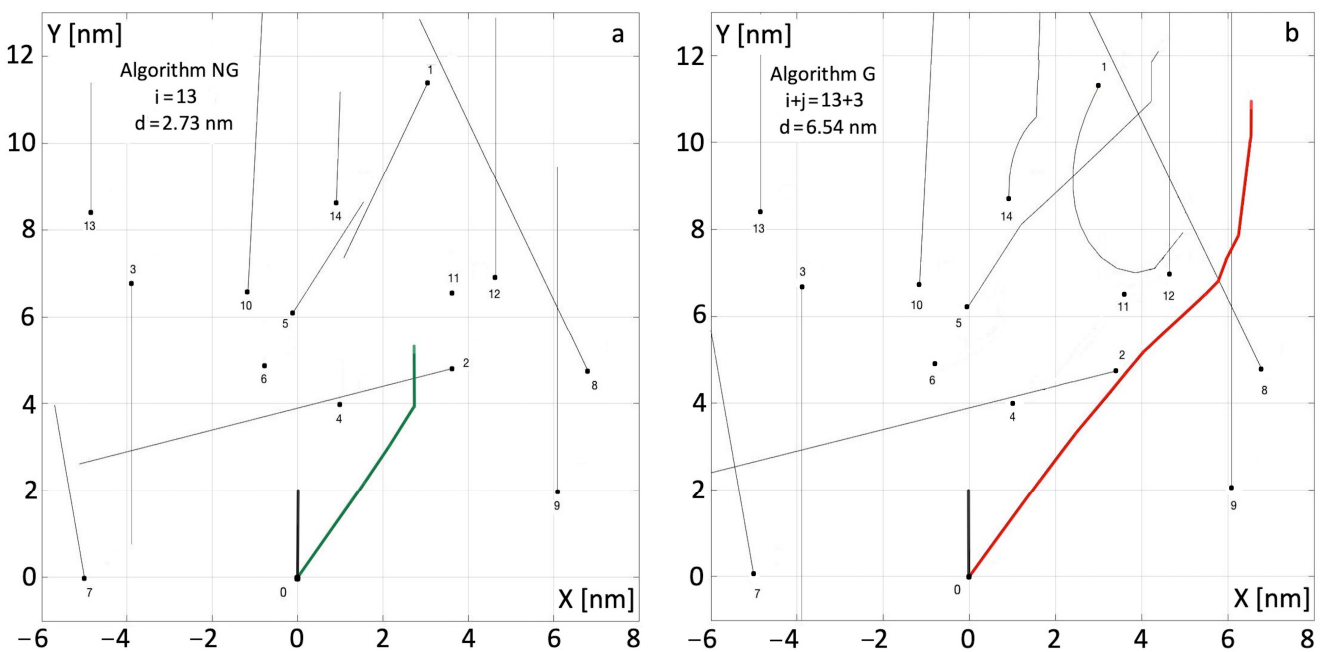


Figure 7. Safe trajectory of our autonomous surface object using set D of objects maneuvering strategies: (a) non-game control (green); (b) game control (red).

The above differences result from the specificity of the G algorithm, which takes into account unforeseen maneuvers of other autonomous surface objects.

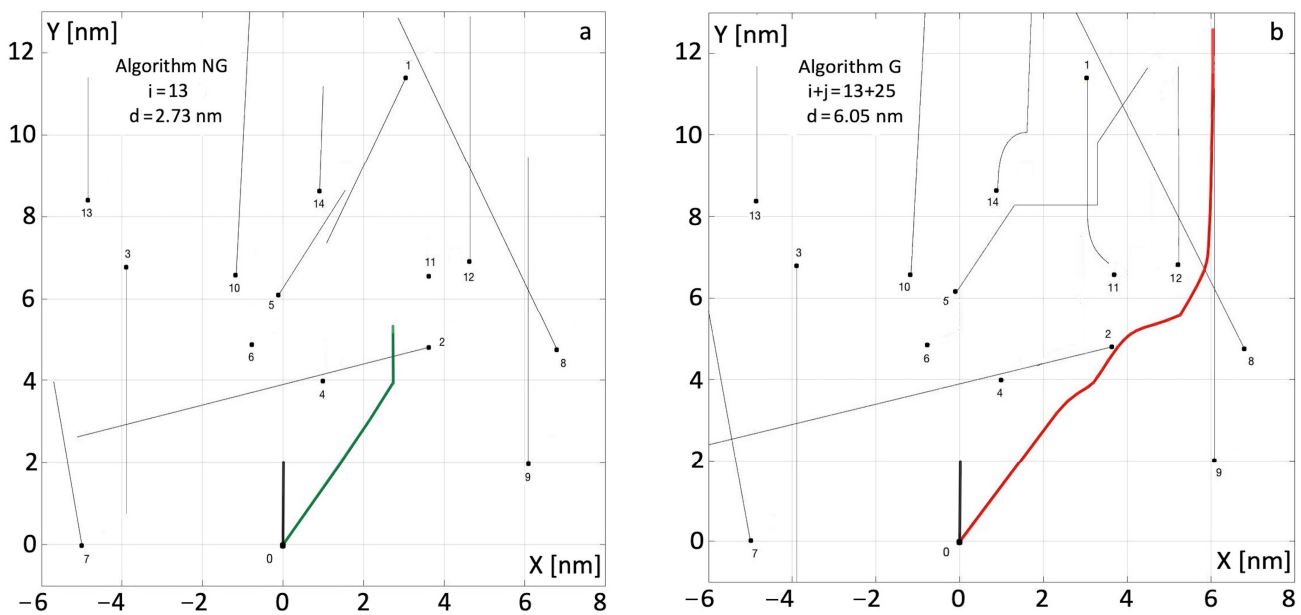


Figure 8. Safe trajectory of our autonomous surface object using set E of objects maneuvering strategies: (a) non-game control (green); (b) game control (red).

4.2. Safe Control Sensitivity

Sensitivity analysis concerns the assessment of the quality of functioning optimal and safe systems control of autonomous surface objects. The sensitivity functions s_x of the optimal and safe control u of the game process described by state variables x can be presented as partial derivatives of the quality control index Q [27–29]:

$$s_x = \frac{\partial Q[x(u)]}{\partial x} \tag{9}$$

where Q is an index of the optimal control quality described by Formula (4).

The sensitivity of the safe control which is treated as the sensitivity s_x of the collision risk r_k to deviations in the measured values x_e of the individual components of the process state from their actual values x is:

$$s_x = \frac{\partial Q_0^k}{\partial x} = \frac{\partial r_k}{\partial x} = \frac{r_k(x_e) - r_k(x)}{r_k(x)} = \left\{ s_{V_0}, s_{\psi_0}, s_{V_k}, s_{\psi_k}, s_{D_k}, s_{N_k}, s_{D_s}, s_{t_s} \right\} \tag{10}$$

where $x(V_0, \psi_0, V_k, \psi_k, D_k, N_k, D_s, t_s, n)$ is the set of real values of information about the process state; t_s is the step time calculation trajectory; $n = i + j$ is the number of admissible strategies of autonomous surface objects; and $x_e(V_0 \pm \delta V_0, \psi_0 \pm \delta \psi_0, V_k \pm \delta V_k, \psi_k \pm \delta \psi_k, D_k \pm \delta D_k, N_k \pm \delta N_k, D_s \pm \delta D_s, t_s \pm \delta t_s, n \pm \delta n)$ is the information about the process state with measurement errors or possibility control.

Figures 9–12 show the sensitivity characteristics of the collision risk to changes in the eight components of the motion of objects and to a different number of acceptable strategies of autonomous surface objects.

Sensitivity analysis of safety control should be conducted in terms of the information inaccuracy obtained from the ARPA anti-collision radar system in the current situation and in relation to changes in the kinematic parameters and dynamic control.

The permissible average errors that may be caused by an anti-collision system sensor have the following values for:

- Log—speed $\delta V_0, \delta V_k: \pm 0.5$ kn;
- Gyrocompass—course $\delta \psi_0, \delta \psi_k: \pm 0.5^\circ$;
- Radar—distance $\delta D_k: \pm 0.05$ nm, bearing $\delta N_k: \pm 0.25^\circ$;

- COLREGs—safe distance δD_s : +100%/−40%, subjective error of the navigator in assessing the situation.

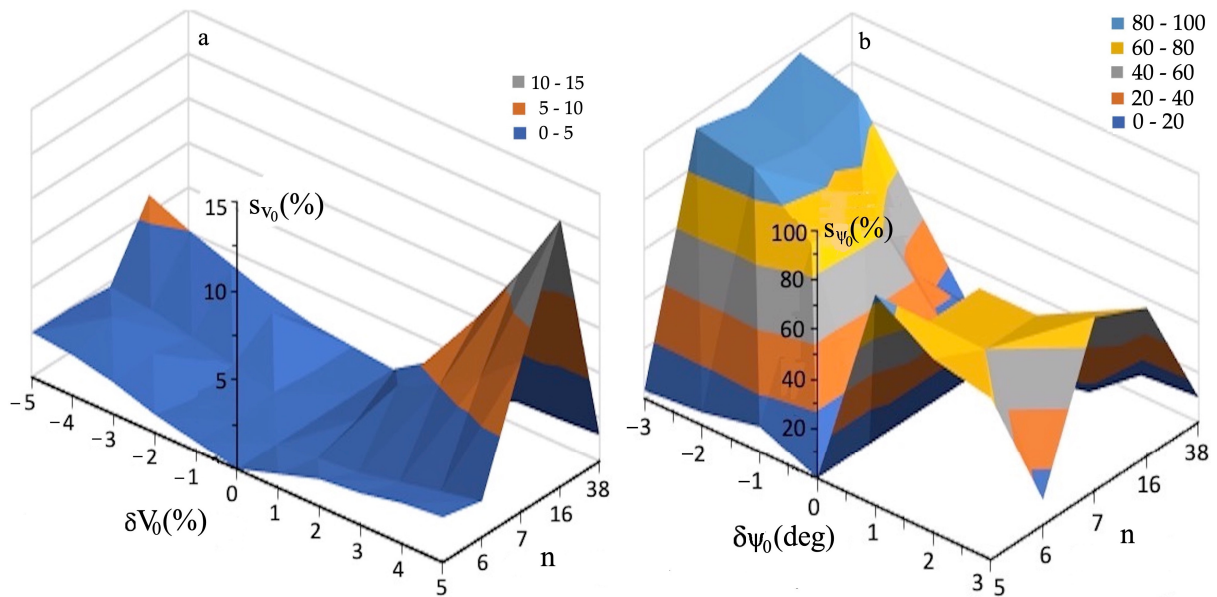


Figure 9. The dependence of the collision risk sensitivity s_{V_0} and s_{ψ_0} on the inaccuracy of the measurement of the: (a) speed δV_0 ; (b) course $\delta \psi_0$ of our autonomous surface 0 object, and the number n of acceptable strategies of autonomous surface objects.

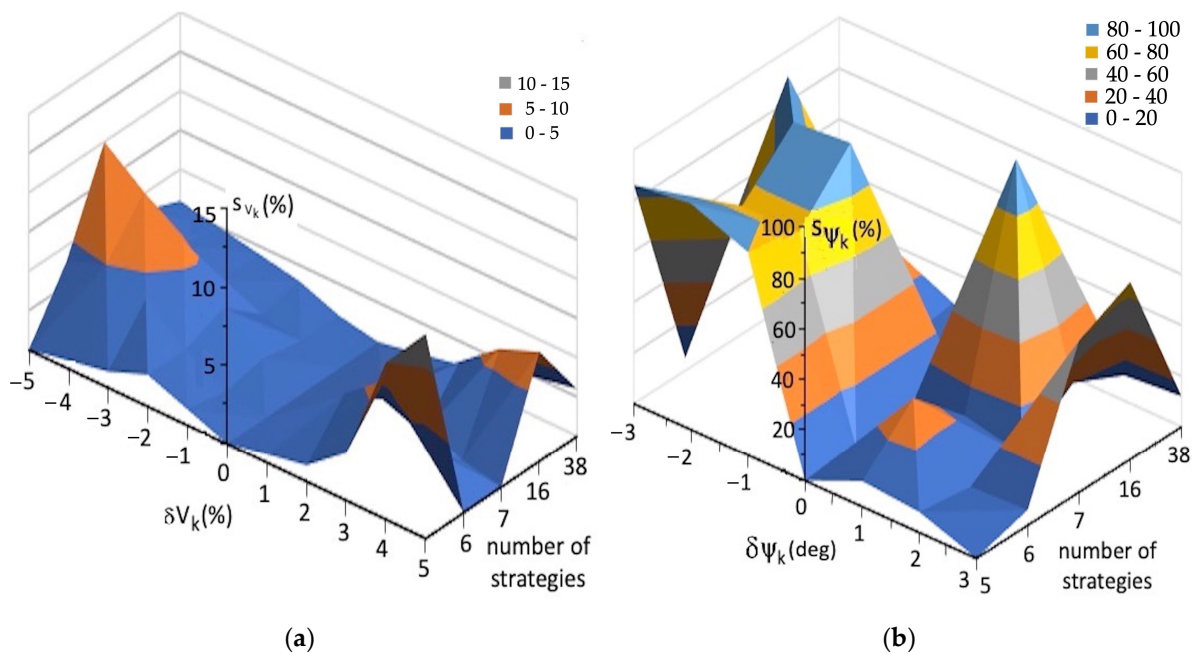


Figure 10. The dependence of the collision risk sensitivity s_{V_k} and s_{ψ_k} on the inaccuracy of the measurement of the: (a) speed δV_k ; (b) course $\delta \psi_k$ of another autonomous surface k object, and the number n of acceptable strategies of autonomous surface objects.

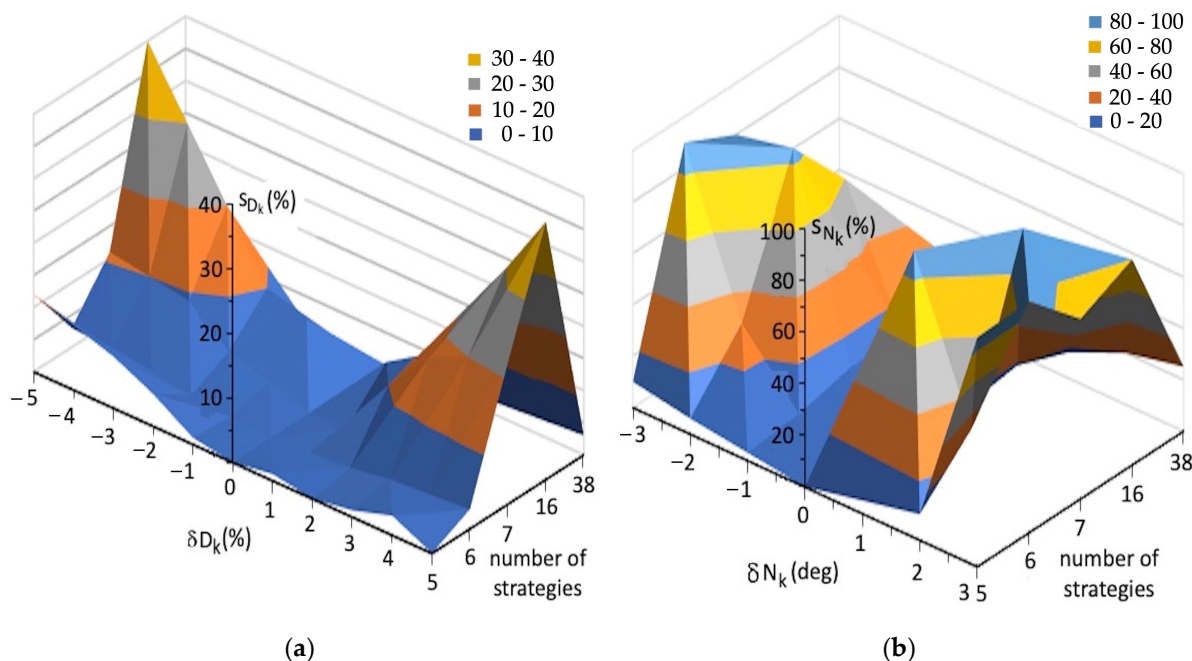


Figure 11. The dependence of the collision risk sensitivity s_{D_k} and s_{N_k} on the inaccuracy of the measurement of the: (a) distance δD_k ; (b) bearing δN_k of another autonomous surface k object, and the number n of acceptable strategies of autonomous surface objects.

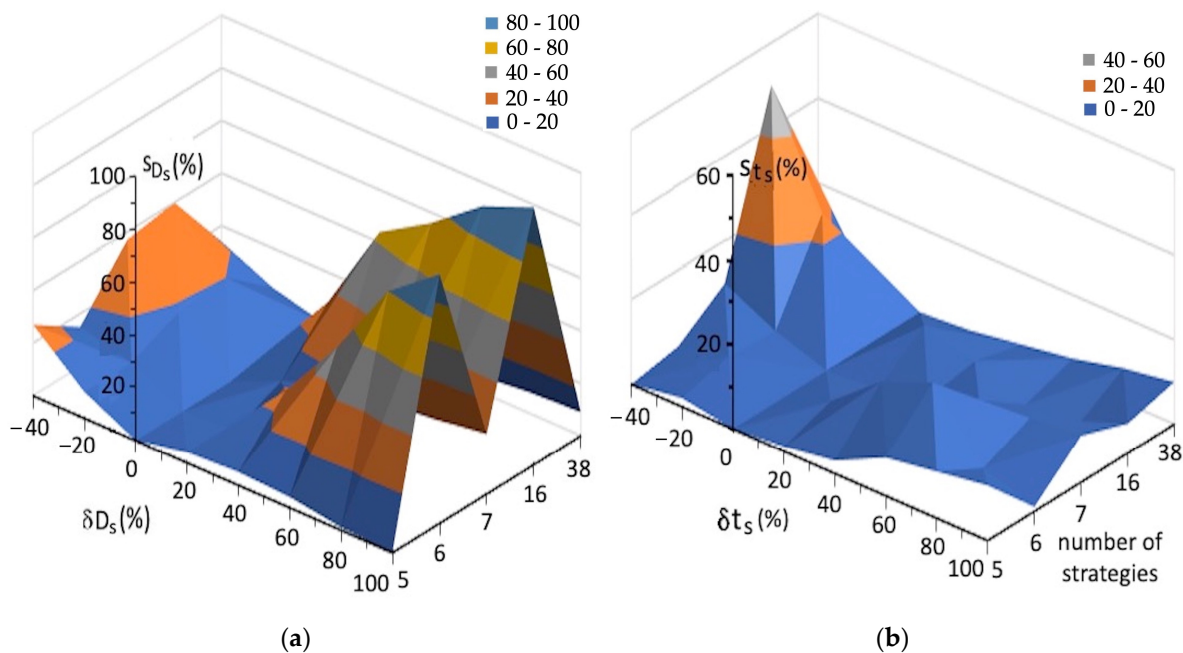


Figure 12. The dependence of the collision risk sensitivity s_{D_s} and s_{t_s} on the inaccuracy of the measurement of the: (a) safe distance δD_s ; (b) step time δt_s of calculations, and the number n of acceptable strategies of autonomous surface objects.

The algebraic sum of all errors affecting the image of the navigational situation cannot exceed $\pm 5\%$ for absolute values and $\pm 3^\circ$ for angular quantities.

The course of the s_V and s_{V_k} characteristics shows that in order to reduce the sensitivity of the safe control by half, the velocity of the objects should be measured with an acceptable error of no more than $\delta V_0 = \delta V_k = 0.2$ kn.

On the other hand, the course of the s_{ψ} and $s_{\psi k}$ characteristics shows that in order to reduce the sensitivity of the safe control by half, the course of the objects should be measured with an acceptable error of no more than $\delta\psi_0 = \delta\psi_k = 0.2$ deg.

The course of the s_D and s_{Dk} characteristics shows that in order to reduce the sensitivity of the safe control by half, the distances of the objects should be measured with an acceptable error of no more than $\delta D_k = 0.02$ nm.

On the other hand, the course of the s_N and s_{Nk} characteristics shows that in order to reduce the sensitivity of the safe control by half, the bearings of the objects should be measured with an acceptable error of no more than $\delta N_k = 0.1$ deg.

The analysis of the sensitivity characteristics of the collision risk of the autonomous surface object's control allows us to draw the following conclusions:

- Sensitivity is at its greatest to measurement errors of angular variables of the process state in the form of the course and bearing;
- Sensitivity increases with increasing traffic safety requirements, defined by the safe distance D_s between objects;
- Sensitivity decreases with an increasing step time t_s value;
- Underestimating the own speed V_0 is better than overestimating it because the risk of collision increases as the speed of the moving object increases;
- Sensitivity decreases with an increase in the number n of acceptable strategies of autonomous surface k objects, which is a positive feature of robust control systems on the impact of any external influences, and results from the possibility of more accurate control with a larger number n of acceptable control strategies.

5. Conclusions

The multi-stage matrix game model enables the synthesis of a computer program for calculating the safe path of an autonomous surface object and other autonomous surface objects that may perform unforeseen maneuvers.

The safe path of an autonomous surface object and its deviation from the initial motion depend on the number of allowed objects strategies.

Collision risk sensitivity characteristics make it possible to determine the required accuracy of measurement of the control process state variables.

This article has not yet solved many of the problems associated with the design, testing and application of new game control systems for autonomous surface objects. Future research should consider:

- Integration of real-time data from the ARPA anti-collision system about changes in the course and speed of other objects;
- Introduction of a mechanism for continuous learning of the control system through the current mapping of the value of the safe passing distance of other objects by an artificial neural network;
- Development of a process model that takes into account non-linear dynamic properties of objects in the differential game form;
- Appropriate semantic interpretation of COLREG requirements;
- More accurate representation of the optimal control process using selected methods of artificial intelligence.

Funding: This research was funded by a research project of the Electrical Engineering Faculty, Gdynia Maritime University, Poland, No. WE/2023/PZ/02: "Control algorithms synthesis of autonomous objects".

Data Availability Statement: Not applicable.

Conflicts of Interest: The author declares no conflict of interest.

Abbreviations

ARPA	Automatic Radar Plotting Aid
ASC	Autonomous Surface Unit
ASV	Autonomous Surface Vehicle
AUV	Autonomous Underwater Vehicle
COLREGs	Collision Regulations
DCPA	Distance to Closest Point of Approach
LQR	Linear Quadratic Regulator
PID	Proportional Integral Differential
TCPA	Time to Closest Point of Approach
USV	Unmanned Surface Vehicle

References

- Zhao, J.; Yan, W.; Jin, X. Brief review of autonomous surface crafts. *ICIC Express Lett.* **2011**, *5*, 4381–4386.
- Barrera, C.; Padron, I.; Luis, F.S.; Llinas, O. Trends and challenges in unmanned surface vehicles (USV): From survey to shipping. *TransNav Int. J. Mar. Navig. Saf. Sea Transp.* **2021**, *15*, 135–142. [[CrossRef](#)]
- Choi, J.; Park, J.; Jung, J.; Lee, Y.; Choi, H.T. Development of an Autonomous Surface Vehicle and Performance Evaluation of Autonomous Navigation Technologies. *Int. J. Control Autom. Syst.* **2020**, *18*, 535–545. [[CrossRef](#)]
- Chen, Y.; Chen, X.; Zhu, J.; Lin, F.; Chen, B.M. Development of an Autonomous Unmanned Surface Vehicle with Object Detection Using Deep Learning. In Proceedings of the IECON 2018–44th Annual Conference of the IEEE Industrial Electronics Society, Washington, DC, USA, 26 December 2018; pp. 5636–5641. [[CrossRef](#)]
- Sneha, T. *The Design and Control of an Economical Autonomous Surface Vehicle for Object Detection*; Princeton University: Princeton, NJ, USA, 2020. Available online: <https://arks.princeton.edu/ark:/88435/dsp01gh93h2567> (accessed on 10 July 2023).
- Omrani, E.; Mousazadeh, H.; Omid, M.; Masouleh, M.T.; Jafarbiglu, H.; Salmani-Zakaria, Y.; Makhsoos, A.; Monhaseri, F.; Kiapei, A. Dynamic and static object detection and tracking in an autonomous surface vehicle. *Ships Offshore Struct.* **2020**, *15*, 711–721. [[CrossRef](#)]
- Zhang, M.; Zhao, D.; Sheng, C.; Liu, Z.; Cai, W. Long-Strip Target Detection and Tracking with Autonomous Surface Vehicle. *J. Mar. Sci. Eng.* **2023**, *11*, 106. [[CrossRef](#)]
- Lee, M.F.R.; Lin, C.Y. Object Tracking for an Autonomous Unmanned Surface Vehicle. *Machines* **2022**, *10*, 378. [[CrossRef](#)]
- Hongguang, L.; Yong, Y. Fast Path Planning for Autonomous Ships in Restricted Waters. *Appl. Sci.* **2018**, *12*, 2592. [[CrossRef](#)]
- Zhou, C.; Gu, S.; Wen, Y.; Du, Z.; Xiao, C.; Huang, L.; Zhu, M. The review unmanned surface vehicle path planning: Based on multi-modality constraint. *Ocean Eng.* **2020**, *200*, 107043. [[CrossRef](#)]
- Du, Z.; Negenborn, R.R.; Reppa, V. Review of floating object manipulation by autonomous multi-vessel systems. *Annu. Rev. Control* **2022**, *55*, 255–278. [[CrossRef](#)]
- Martins, A.; Almeida, J.M.; Ferreira, H.; Silva, H.; Dias, N.; Dias, A.; Almeida, C.; Silva, E.P. Autonomous Surface Vehicle Docking Manoeuvre with Visual Information. In Proceedings of the IEEE International Conference on Robotics and Automation, Roma, Italy, 10–14 April 2007. [[CrossRef](#)]
- Park, H.; Ham, S.H.; Kim, T.; An, D. Object Recognition and Tracking in Moving Videos for Maritime Autonomous Surface Ships. *J. Mar. Sci. Eng.* **2022**, *10*, 841. [[CrossRef](#)]
- Li, J.; Zhang, G.; Shan, Q.; Zhang, W. A Novel Cooperative Design for USV-UAV Systems: 3D Mapping Guidance and Adaptive Fuzzy Control. *IEEE Trans. Control. Netw. Syst.* **2022**, *11*, 564–574. [[CrossRef](#)]
- Wang, W.; Gheneti, B.; Mateos, L.A.; Duarte, F.; Ratti, C.; Rus, D. Roboat: An Autonomous Surface Vehicle for Urban Waterways. In Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Macau, China, 3–8 November 2019; pp. 6340–6347. [[CrossRef](#)]
- Hongguang, L.; Yong, Y. COLREGS-Constrained Real-time Path Planning for Autonomous Ships Using Modified Artificial Potential Fields. *J. Navig.* **2018**, *72*, 588–608. [[CrossRef](#)]
- Kim, H.G.; Yun, S.J.; Choi, G.H.; Ryu, J.K.; Suh, J.H. Collision Avoidance Algorithm Based on COLREGs for Unmanned Surface Vehicle. *J. Mar. Sci. Eng.* **2021**, *9*, 863. [[CrossRef](#)]
- Zhong, S.; Wen, Y.; Huang, Y.; Cheng, X.; Huang, L. Ontological Ship Behavior Modeling Based COLREGs for Knowledge Reasoning. *J. Mar. Sci. Eng.* **2022**, *10*, 203. [[CrossRef](#)]
- Hu, L.; Hu, H.; Naeem, W.; Wang, Z. A review on COLREGs-compliant navigation of autonomous surface vehicles: From traditional to learning-based approaches. *J. Autom. Intell.* **2022**, *1*, 100003. [[CrossRef](#)]
- Sun, Z.; Sun, H.; Li, P.; Zou, J. Self-organizing cooperative pursuit strategy for multi-USV with dynamic obstacle ships. *J. Mar. Sci. Eng.* **2022**, *10*, 562. [[CrossRef](#)]
- Koznowski, W.; Kula, K.; Lazarowska, A.; Lisowski, J.; Miller, A.; Rak, A.; Rybczak, M.; Mohamed-Seghir, M.; Tomera, M. Research on Synthesis of Multi-Layer Intelligent System for Optimal and Safe Control of Marine Autonomous Object. *Electronics* **2023**, *12*, 3299. [[CrossRef](#)]

22. Walker, K.L.; Gabi, R.; Aracri, S.; Cao, Y.; Stokes, A.A.; Kiprakis, A.; Serchi, F.G. Experimental Validation of Wave Induced Disturbances for Predictive Station Keeping of a Remotely Operated Vehicle. *IEEE Robot. Autom. Lett.* **2021**, *6*, 5421–5428. [[CrossRef](#)]
23. Gabi, R.; Davey, T.; Cao, Y.; Li, Q.; Li, B.; Walker, K.L.; Serchi, F.G.; Aracri, S.; Kiprakis, A.; Stokes, A.A.; et al. Experimental Force Data of a Restrained ROV under Waves and Current. *Data* **2020**, *5*, 57. [[CrossRef](#)]
24. Lisowski, J. Synthesis of a Path-Planning Algorithm for Autonomous Robots Moving in a Game Environment during Collision Avoidance. *Electronics* **2021**, *10*, 675. [[CrossRef](#)]
25. Lisowski, J. Optimization Methods in Maritime Transport and Logistics. *Pol. Marit. Res.* **2018**, *25*, 30–38. [[CrossRef](#)]
26. Lisowski, J. The dynamic game models of safe navigation. *TransNav Int. J. Mar. Navig. Saf. Sea Transp.* **2007**, *1*, 11–18. [[CrossRef](#)]
27. Lisowski, J. Sensitivity of computer support game algorithms of a safe ship control. *Int. J. Appl. Math. Comput. Sci.* **2013**, *23*, 439–446. [[CrossRef](#)]
28. Eslami, M. *Theory of Sensitivity in Dynamic Systems*; Springer: Berlin/Heidelberg, Germany, 1994; ISBN 978-3-662-01632-9.
29. Rosenwasser, E.; Yusupov, R. *Sensitivity of Automatic Control Systems*; CRC Press: Boca Raton, FL, USA, 2000; ISBN 978-0-849-322929-8.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.