Control and Cybernetics

vol. **42** (2013) No. 4

Optimal and game control algorithms of a ship in collision situations at sea*

by

Józef Lisowski

Gdynia Maritime University,
Electrical Engineering Faculty,
Department of Ship Automation
Gdynia, Poland
jlis@am.gdynia.pl

Abstract: This paper presents an application of selected methods of optimal and game control theory to determine own ship safe trajectory when passing other ships encountered in good and in restricted visibility at sea. Five algorithms for determining safe trajectory of the own ship in a collision risk situation: non-cooperative positional game, non-cooperative matrix game, cooperative positional game, dynamic optimization, and kinematic optimization are compared. The analysis is illustrated with examples of computer simulations of the algorithms to determine safe and optimal own ship trajectories in the real navigational situations at sea.

Keywords: marine navigation, safety at sea, safe ship control, optimal control, game theory

1. Introduction

The important issues of the theory of decision-making processes in marine navigation should include safe control of the ship in the circumstances shown in Fig. 1.

The process of handling a ship as a multidimensional dynamic object depends both on the accuracy of details referring to the current navigational situation obtained from the Automatic Radar Plotting Aids (ARPA) anti-collision system and on the form of the process model used for control synthesis. The ARPA system ensures monitoring of j encountered ships, determining their movement parameters (speed V_j, course ψ_j) and elements of approaching the own ship moving at a speed V and course ψ (to satisfy $D^j_{\min} = DCPA_j$ - Distance of the Closest Point of Approach, and $T^j_{\min} = TCPA_j$ - Time to the Closest Point of Approach) allowing also for assessing the risk of collision r_j (Bole, Dineley and Wall, 2006).

*Submitted: June 2013; Accepted: September 2013.

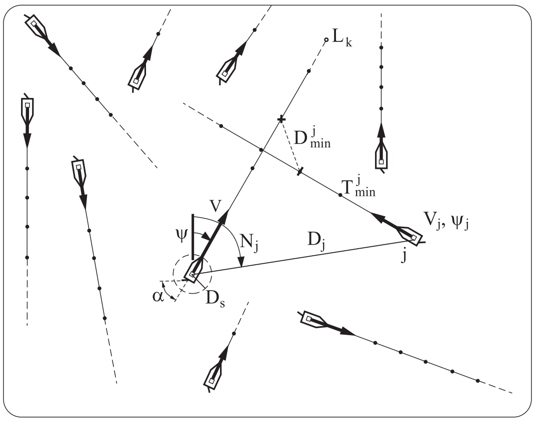

Figure 1. Navigational situation of the own ship passing j encountered ships

In order to ensure the safety of navigation, the ships are obliged to comply with the International Regulations for Preventing Collisions at Sea (COLREG). The COLREG rules distinguish between good and restricted visibility. Under good visibility the rules oblige the own ship to give way to the ship approaching it from Starboard Side. If there are more ships moving in the vicinity, at first, it is necessary to find the most hazardous ship and adapt the COLREG rules to it, and then to verify the determined anti-collision manoeuvre in relation to the remaining ships. In practice, when determining the safe manoeuvre under good visibility conditions, the assumed safe passing distance value D_s is from 0.1 to 1.0 nautical mile, depending upon the ships sizes and their respective speeds. Under restricted visibility conditions the rule of giving way to the ship on Starboard Side is not obligatory. The rule 19 in COLREG states that every vessel should proceed at a safe speed adapted to the prevailing circumstances and restricted visibility. So, in practice, when determining the safe manoeuvre under restricted visibility conditions, the assumed safe passing distance value D_s is from 1.0 to 3.0 nautical miles (Bist, 2000; Cockcroft and Lameijer, 2006).

The actual process of a ship passing other objects very often occurs under conditions of uncertainty and conflict accompanied by an inadequate co-operation between the ships within COLREG. It is, therefore, reasonable to investigate, develop and represent methods of a ship safe handling, applying the rules of a theory based on dynamic game and optimization (Cahill, 2002; Engwerda, 2005; Gluver and Olsen, 1998). The model of the process consists of the kinematics and the dynamics of the ship movement, the disturbances, the strategies of the encountered ships and the quality control index of the own ship. There are various methods for the avoidance of ship collision. The simplest method is to determine the manoeuvre of change in the course or the speed of own ship in relation to the most dangerous ship encountered (Cymbal, Burmaka and Tupikov, 2007). A more effective method is to determine the safe trajectory of the ship (Hasegawa, Shigemori and Ichiyama, 2000; Lee and Rhee, 2001; Pietrzykowski, 2011; Seghir, 2012; Szlapczynski and Smierzchalski, 2009). Most adequate to the real character of the control process is determination of a game-based trajectory of the ship. The concept of a game-based trajectory describes the own ship trajectory so as to determine which manoeuvres of the encountered ships are taken into consideration (Lisowski, 2013).

The diversity of the possible models directly affects the synthesis of the ship control algorithms, which are, afterwards, affected by the ship control device, directly linked to the ARPA system, and consequently determines the effects of safe and optimal control. Fig. 2 represents a set of compromises of a ship safety control measured in terms of a collision risk and time-optimal strategy of the own ship control.

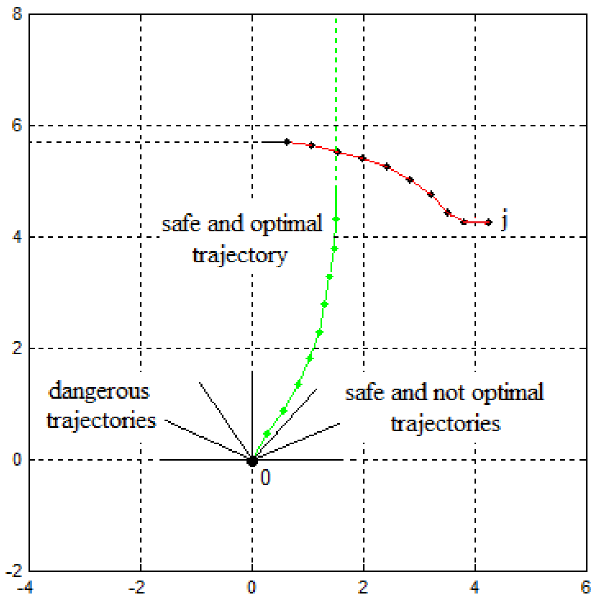

Figure 2. Possible trajectories of own ship 0 in situation of passing encountered ship j

2. Game and optimal models of safe ship control

The way of controlling a ship, which is a multi-dimensional and non-linear dynamic object, depends on the scope and accuracy of information on the prevailing navigational situation and on the adopted model of the process. The variety of models to be potentially adopted directly influences the synthesis of various methods supporting the navigator's task, and then bears on the effects of the safe control of the own ship movement. One should distinguish between the exact model as the basic one and the approximate models that are used for the synthesis of control programs. Fig. 3 shows the block diagram of the possible sequence of models for control process, where: $\vec{U}$– vector of the own ship control, $\vec{U}_j$ – control vector of the $j-$th ship, $\vec{X}_j$ – state vector of the $j-$th ship, $\vec{Z}$ – disturbance vector, $\vec{X}$ – state vector of the process, $\vec{R}$ – vector of safety constraints.

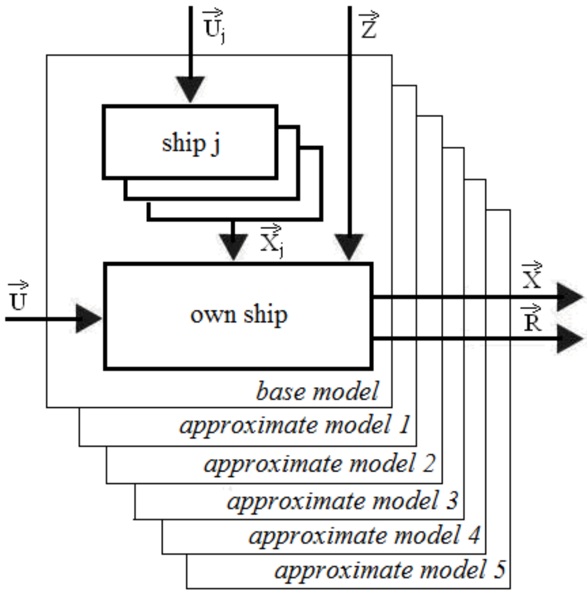

Figure 3. Block diagram of the base and approximate models for control process

2.1. Base differential game model

The most general description of the own ship passing j other encountered ships is the model of a differential game of j moving control objects (Fig. 4).

The properties of the process are described by the state equation:

$$\dot{x}_i = f_i\left(x_{j,\vartheta_j}, u_{j,v_j}, t\right), \quad j = 1, 2, ...m \tag{1}$$

where

$\vec{x}_{0,\theta_0}(t)$ – θ_0– dimensional vector of the process state of the own ship determined within the time span $t \in [t_0, t_k]$;

$\vec{x}_{j,\theta_j}(t)$ – θ_j– dimensional vector of the process state for the $j-$th encountered ship;

$\vec{u}_{0,\nu_0}(t)$ – ν_0– dimensional control vector of the own ship;

$\vec{u}_{j,\nu_j}(t)$ – ν_j – dimensional control vector of the j-th encountered ship;

m – number of ships other than the own ship in the analysed area.

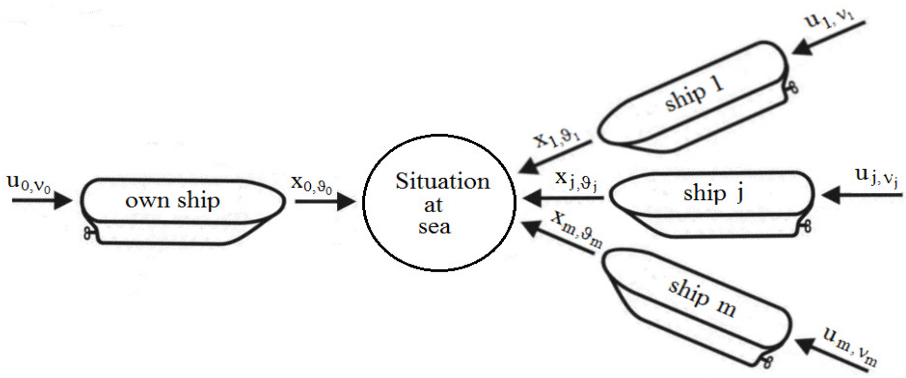

Figure 4. Block diagram of the basic model of differential game

The constraints of the control and the state of the process are connected with the basic condition for the safe passing of the ships at a safe distance $D_s$in compliance with the rules in COLREG, generally in the following form:

$$g_j(x_{j,\theta_j}, u_{j,v_j}) \geq 0. \tag{2}$$

Goal function has a form of the payments – the integral payment and the final one:

$$I_{0,j} = \int_{t_0}^{t_k} x^2_{0,\theta_0}(t)dt + r_j(t_k) + d(t_k) \rightarrow min \tag{3}$$

In equation (3) the first part - integral payment represents additional distance covered by the own ship while passing the encountered ships and the second part - represents the final payment, which determines the final risk of collision $r_j(t_k)$ relative to the $j-$th ship and final deviation of the own ship $d(t_k)$ from the reference trajectory (Osborne, 2004; Isaacs, 1965; Millington and Funge, 2009; Baba and Jain, 2001; Nisan, Roughgarden, Tardos and Vazirani, 2007).

2.2. Approximate models

Gradually, in the framework of the basic model description of kinematics and dynamics of the traffic, approximate models are obtained, which can be ordered from the most complex, such as non-cooperative positional game to the simplest, such as the kinematic optimization model.

Thus, for the practical synthesis of safe control methods various simplified models are formulated:

- non-cooperative positional game,
- non-cooperative matrix game,
- cooperative positional game,

- dynamic optimization,
- kinematic optimization.

The concepts of cooperation and non-cooperation refer to the degree of cooperation between the ship manoeuvres when passing pursuant to the rules in COLREG.

The degree of model simplification depends on the optimal control method applied and the level of cooperation between the ships. For each of the five simplified models, a method has been developed or adopted for determining the optimal and safe trajectory of the own ship in the collision situation. Table 1 shows the assignment of appropriate control algorithms to the respective models.

Table 1. Methods of determining own ship strategies in a collision situation

Algorithm	Approximate model	Method of optimization	Form of trajectory
NPG	non-cooperative positional game	linear programming	game trajectory
NMG	non-cooperative matrix game	linear programming	game trajectory
CPG	cooperative positional game	linear programming	game trajectory
DO	dynamic optimization	dynamic programming	optimal trajectory
KO	kinematic optimization	linear programming	optimal trajectory

3. Computer supported algorithms

In practice, methods of selecting a safe trajectory assume a form of control algorithms supporting the navigator's decision in a collision situation. Algorithms are programmed in the memory of a Programmable Logic Controller PLC. This generates an option within the ARPA anti-collision system or a training simulator (Fig. 5).

3.1. Algorithm of non-cooperative positional game

The control objective of the own ship is to avoid collision with approaching ships, which for various reasons are on the collision course. These reasons may be: delay or failure in deciding on a maneuver, misrepresentation of the COLREG rules, failure of navigation equipment, difficult weather conditions. This situation is best described by the mathematical model of the non-cooperative positional game. The objective function of control of the own ship corresponds to the minimum loss associated with safe overtaking of the ships encountered.

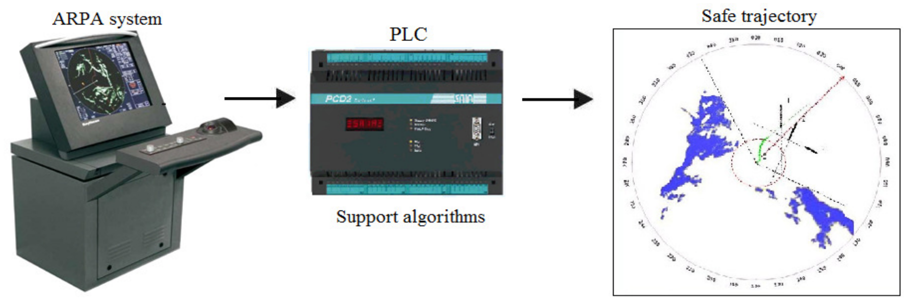

Figure 5. Structure of a computer supported system of navigator's decision in a collision situation

The concept of positional game refers to the type of game where the ship strategies depend upon the ship positions. The optimal control of the own ship, $u_0^*(t)$, which is equivalent to the optimal positional steering, $u_0^*(p)$ for the current position $p(t)$, is determined from the condition:

$$I^* = \min_{u_0 \in U_0 = \bigcap_{j=1}^{m} U_0^j} \left(\max_{u_j^m \in U_j} \min_{u_0^j \in U_0^j(u_j)} \int_{t_0}^{t_k} u_0(t)\ dt \right) = S_0^*(x_0, L_k)\ . \quad (4)$$

The value S_0 refers to the objective function of control of the own ship, characterising the distance of the ship at the initial moment t_0 to the nearest turning point L_k on the reference $p_r(t)$ route of the voyage.

Formula (4), which leads to determination of the safe control of a ship, implying optimum values for the respective objective function of control, makes it dependent on the form of the model of control process.

The optimal control of the own ship is calculated at each discrete stage of the ship movement by applying the SIMPLEX method to solve the problem of the linear programming, assuming the relationship (4) as the goal function and the control constraints (2). Using the function of *lp – linear programming* from the Optimization Toolbox MATLAB, the non-cooperative positional game manoeuvring NPG program has been designed for the determination of the own ship safe trajectory in a collision situation.

3.2. Algorithm of non-cooperative matrix game

The control objective is the same as in the NPG algorithm, but another objective function of control of the own ship is applied, taking the form of collision risk. The general dynamic non-linear model of ship collision at sea is simplified to the kinematic linear model, which can be introduced in the form of a matrix of collision risk between own ship and encountered ships (Modarres, 2006; Zio, 2009).

The matrix game $\boldsymbol{R}= [r_j(\nu_j,\ \nu_0)]$ includes the value of a collision risk r_j with regard to the determined strategies ν_0 of the own ship and those ν_j of the $j-$th encountered ship. The collision risk r_j of the own ship in relation to the $j-$th ship encountered, assuming the values from 0 to 1, is represented by the degree of hazard resulting from the current values of minimum distances and time of approaching the encountered ship in relation to their safe values. The value of risk of collision, r_j, is defined by referring the current situation of the approach, described by the parameters $D^j_{\min}$ and $T^j_{\min}$, to the assumed assessment of the situation, defined as safe and determined by the safe distance of approach D_s and the time to reach a safe passing distance T_s- that are necessary to execute a manoeuvre avoiding a collision with regard to actual distance D_j between own ship and encountered $j-$th ship:

$$r_j = \frac{1}{\sqrt{\lambda_d\left(\frac{D^j_{\min}}{D_s}\right)^2 + \lambda_t\left(\frac{T^j_{\min}}{T_s}\right)^2 + \left(\frac{D_j}{D_s}\right)^2}}. \tag{5}$$

The coefficients in formula (5), λ_d and λ_t, are relative to: the state of the visibility at sea, the dynamic length and the beam, the kind of marine area - open or limited.

As a result of applying the following form for the control goal:

$$\left(I^j_0\right)^* = \min_{\nu_0} \ \max_{\nu_j} r_j \tag{6}$$

the probability matrix $\boldsymbol{P}= p_j\left(\nu_j, \nu_0\right)$ of using the particular pure strategies may be obtained. The solution for the control goal is the strategy with the highest probability:

$$\left(u^{\nu_0}_0\right)^* = u^{\nu_0}_0\left(max\ p_j\left(\nu_j, \nu_0\right)\right). \tag{7}$$

Using the function of *lp – linear programming* from the Optimization Toolbox MATLAB, the matrix game non-cooperative manoeuvring NMG algorithm has been designed for the determination of the own ship safe trajectory in a collision situation.

3.3. Algorithm of cooperative positional game

The control objective of the own ship is to avoid collision with the approaching ships, which, in this particular case, cooperate in passing at a safe distance, according to the rules in COLREG. The objective function of control of own ship corresponds to the minimum loss associated with safe passing by the ships encountered.

The goal function (4) for the cooperative game has the form:

$$I = \min_{u_0 \in U_0 = \bigcap\limits_{j=1}^{m} U^j_0} \left(\min_{u^m_j \in U_j} \ \min_{u^j_0 \in U^j_0(u_j)} \int\limits_{t_0}^{t_k} u_0(t)dt \right) = S^*_0(x_0, L_k). \tag{8}$$

3.4. The method of dynamic optimization

The control objective of the own ship is to avoid collision with the approaching ships, which represent, now, the moving constraints, with their shape and size being associated directly with the risk of collision. The objective function of control of own ship corresponds to the minimum time for safe passing by the ships encountered.

Own ship dynamics is represented by the state equations in a discrete form:

$$x_{i,k+1} = x_{i,k} + \Delta x_{i,k}(x_i, u_1, u_2) \quad i = 1, 2, ..., 7 \tag{9}$$

where

$x_1 = X_0, x_2 = Y_0, x_3 = \psi, x_4 = \dot{\psi}_{\max}, x_5 = V, x_6 = \dot{V}, x_7 = t$
$u_1 = \alpha_r/\alpha_{\max}, u_2 = n_r/n_{max}$.

The basic criterion for the ship control is to ensure safe passing of the objects, which is considered in the state constraints:

$$g_j(X_j, Y_j,\ t) \leqslant 0. \tag{10}$$

This dependence is determined by the area referred to as the *ship domain* of the collision hazard that assumes the form of a circle, parable, ellipse or hexagon. The ship domains may have permanent or variable shapes generated, for example, by Neural Network Toolbox MATLAB.

Moreover, a criterion of optimization is taken into consideration in the form of smallest possible route loss resulting from the safe passing of the objects, which, at a constant speed of the own ship, leads to the time-optimal control:

$$I(u_1, u_2) = \int\limits_0^{t_k} x_5\, dt \cong x_5 \int\limits_0^{t_k} dt \ \rightarrow\ \min\,. \tag{11}$$

Determination of the optimal control of the ship in terms of an adopted control quality index may be performed by applying Bellman's dynamic programming principle. The optimal time for the ship to go through k stages is as follows:

$$t_k^* = \min_{u_{1,k-2}, u_{2,k-2}} [t_{k-1}^* + \ \Delta\, t_k(x_{1,k}, x_{2,k}, x_{1,k+1},\, x_{2,k+1},\, x_{5,k})] \ . \tag{12}$$

The optimal time for the ship to go through the k stages is a function of the system state at the end of the $k-1$ stage and control $(u_{1,k2}, u_{2,k2})$ at the $k-2$ stage. By going from the first stage to the last one the formula (12) determines the Bellman's functional equation for the process of the ship control by altering the rudder angle and the rotational speed of the screw propeller. The constraints for the state variables and the control values are generated by the *NEUROCONSTR* procedure in the dynamic optimal control DO algorithm for the determination of the own ship safe trajectory in a collision situation.

3.5. The method of kinematic optimization

The control objective of the own ship is to avoid collision with the approaching ships, which do not maneuver and move with a constant course and constant speed. The objective function of control of own ship corresponds to the minimum loss associated with safe overtaking of the ships encountered.

Goal function (4) for kinematic optimization has the form:

$$I = \min_{u_0 \in\ U_0 = \bigcap_{j=1}^{m} U_0^j} \int_{t_0}^{t_k} u_0(t)\ dt \ = \ S_0^* \left(x_0,\ L_k \right) . \tag{13}$$

The optimal trajectory of the own ship allows for the lowest losses of distance to be covered for safe passing of the encountered ships (DO algorithm) or the lowest final deviation from the set up trajectory (NPG, NMG, CPG, KO algorithms).

All algorithms ensure observing the COLREG rules by formulation of appropriate logic functions representing semantic interpretation of manoeuvre diagrams developed by A. G. Corbet, S. H. Hollingdale, E. S. Calvert and K. D. Jones. Each particular type of situation involving the approach of ships is assigned the logical variable value equal to one (Starboard Side manoeuvre) or minus one (Port Side manoeuvre). The use of these algorithms by the own ship does not depend on whether other ships use the same software or not.

4. Computer simulations

Computer simulations of the NPG, NMG, CPG, DO and KO algorithms were carried out in MATLAB/SIMULINK software on the examples of real navigational situations of passing j =9 and j =47 encountered ships in good visibility when $D_s = 1$ *nm* (*nautical miles*). The algorithm for good visibility may be changed into the algorithm for restricted visibility by increasing the value of the safe approach distance D_s.

The stage interval of simulation was equal to 60 through 720 seconds, while the simulation interval is the time period needed for passing the encountered ships and to achieve the final risk rate $r_j = 0$. The ship manoeuvring model represents the ship dynamics model described by the first order inertia with a delay, approximated by manoeuvre anticipating time, change of the course or speed, which was taken to last from 180 to 720 seconds. Grid formation depends on the previously determined distance of initiation of tracking of the encountered ship by the ARPA collision avoidance tracking system (Basar and Olsder, 1998).

4.1. Situation j=9

The results of computer simulations of five algorithms described in the paper are shown in Figs. 6-11.

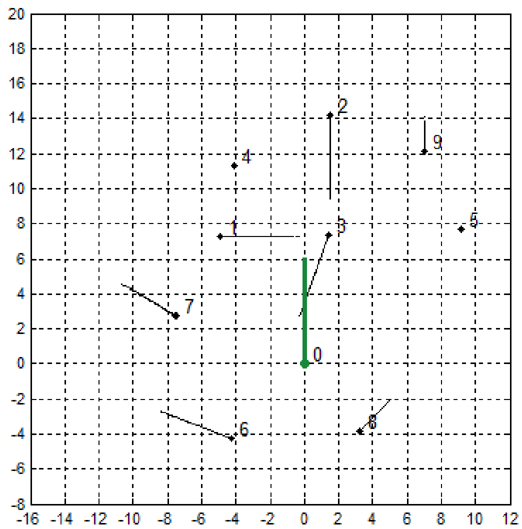

Figure 6. The 18-minute speed vectors of own ship and j=9 encountered ships

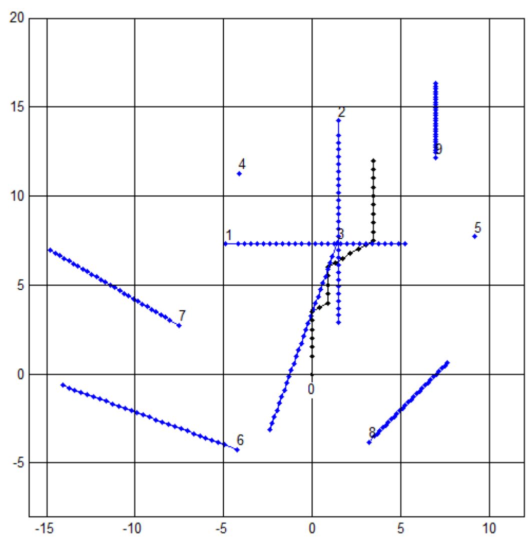

Figure 7. The safe trajectory of own ship for the NPG algorithm in a situation of passing j=9 encountered ships, $r(t_k)$=0, $d(t_k)$=5.19 nm

The NPG algorithm takes into account the lack of cooperation of ships in complying with the rules in COLREG, and is characterized by the greatest deviation from the planned route, in comparison to the remaining methods.

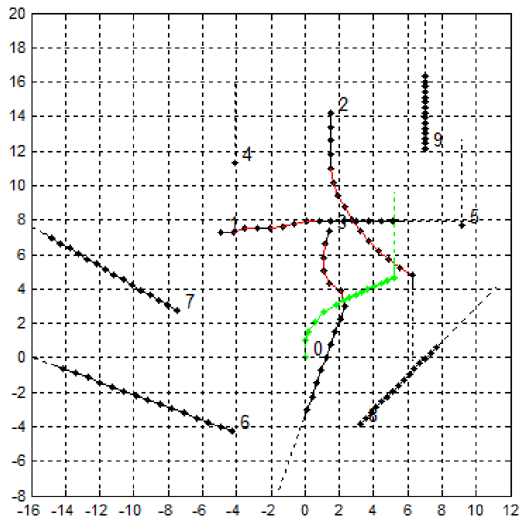

Figure 8. The safe trajectory of the own ship for the NMG algorithm in a situation of passing j=9 encountered ships, $r(t_k)$=0, $d(t_k)$=4.00 nm

The NMG algorithm takes into account the risk of collision resulting from the interpretative error of the rules in COLREG and the lack of cooperation of ships, and is characterized with a smaller deviation from the planned route, but also a higher number of course changes.

The CPG algorithm takes into account the cooperation of ships in compliance with COLREG, which leads to a smaller deviation from the planned route and the shorter time of manoeuvering for achievement of safe distance of the passage.

The DO algorithm, which takes into account the subjectivity of the navigator in the estimation of collision situation across generated domains of the collision risk, assures both small deviation and achievement of the minimum safety distance to encountered ships at the closest point.

The KO algorithm does not take into account the manoeuvering of other ships and constitutes only an improvement over the ARPA system within its domain of realization of the function TRIAL MANOEUVRE, allowing for the detarmination of sequence of anti-collision manoeuvres instead of the single manoeuvre of the own ship.

Thus, Fig. 12 shows a comparison of safe trajectories own ship, as designed with five algorithms: NPG, NMG, CPG, DO and KO.

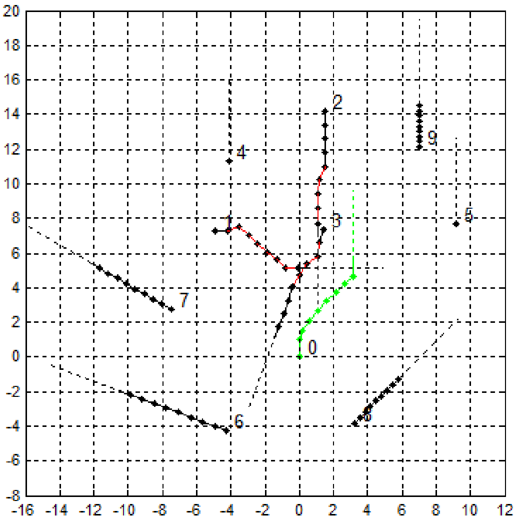

Figure 9. The safe trajectory of own ship for the CPG algorithm in a situation of passing j=9 encountered ships, $r(t_k)$=0, $d(t_k)$=3.15 nm

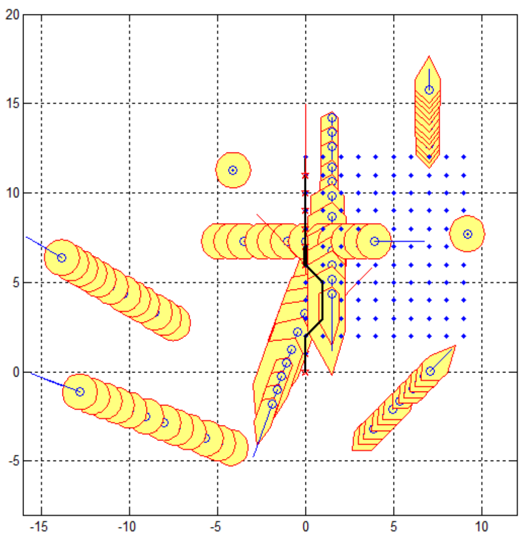

Figure 10. The safe trajectory of own ship for the DO algorithm in a situation of passing j=9 encountered ships, t_K^*=0.66h

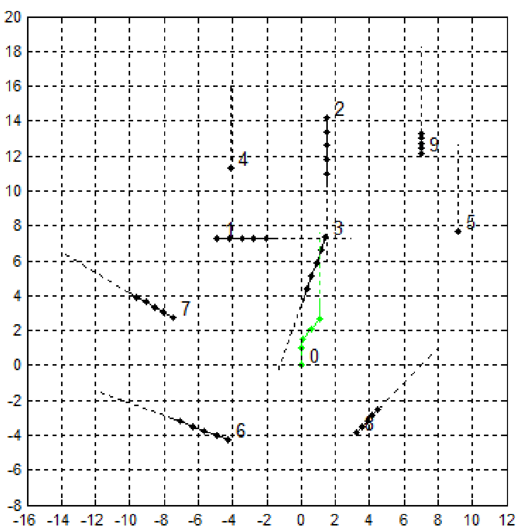

Figure 11. The safe trajectory of the own ship for the KO algorithm in a situation of passing j=9 encountered ships, $r(t_k)$=0, $d(t_k)$=1.08 nm

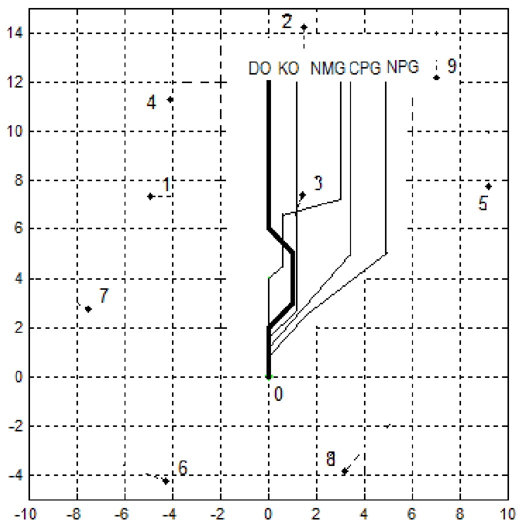

Figure 12. The comparison of safe trajectories own ship in the situation of j=9 passing ships, designed with five algorithms: NPG, NMG, CPG, DO and KO

4.2. Situation j=47

The corresponding results of computer simulations for the five algorithms described in the paper are shown in Figs. 13-19.

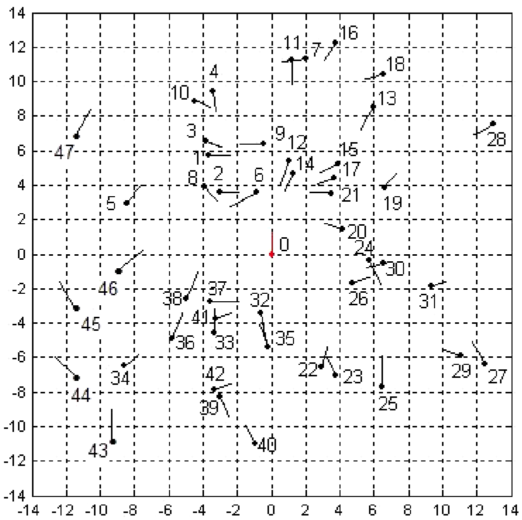

Figure 13. The 6-minute speed vectors of own ship and j=47 encountered ships

The comparison shows that the algorithm of dynamic optimization DO, using hundreds of navigators of pre-trained artificial neural network to generate the domains of the encountered ships, allows for determining the safe trajectory of own ship in the collision situation with the smallest deviation from the reference trajectory and for accounting for the subjectivity of the navigator.

5. Conclusions

The control algorithms considered in this paper are, in a certain sense, formal models for the thinking processes of a navigating officer steering an own ship. The developed algorithms take into consideration the COLREG Rules, the advance time of the manoeuvre, resulting from the approximation of the ship dynamic properties, and allow for evaluating the final deviation of the real trajectory from the reference value.

These algorithms can be used for computer support of the navigator's safe manoeuvring decision in collision situations using information from ARPA anti-collision radar system.

The use of these algorithms by the own ship is not conditioned by the fact whether other ships use the same software. NPG, CPG and NMG game algorithms take into account in the decision making the changes of course and speed of other ships, whether cooperating or not cooperating in accordance with the

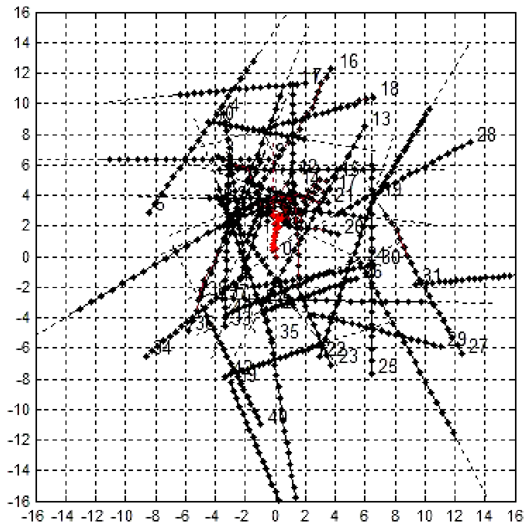

Figure 14. The safe trajectory of own ship for the NPG algorithm in a situation of passing j=47 encountered ships, $r(t_k)$=0, $d(t_k)$=0.13 nm

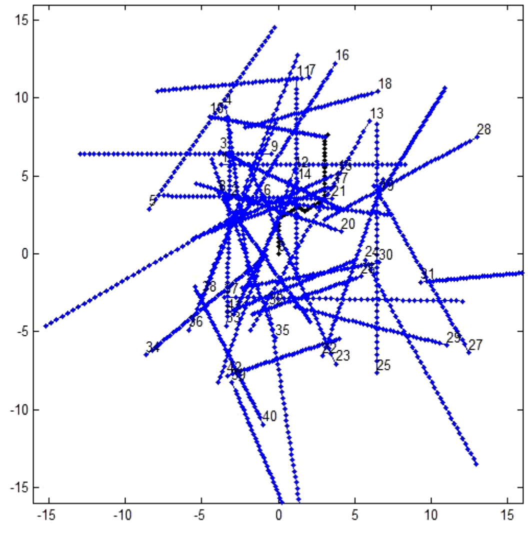

Figure 15. The safe trajectory of the own ship for the NMG algorithm in a situation of passing j=47 encountered ships, $r(t_k)$=0, $d(t_k)$=3.67 nm

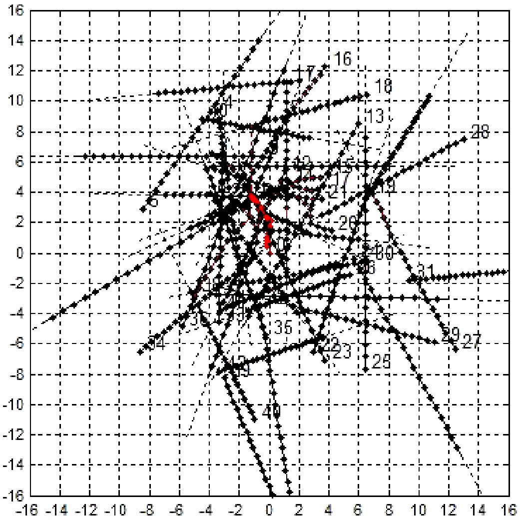

Figure 16. The safe trajectory of own ship for the CPG algorithm in a situation of passing j=47 encountered ships, $r(t_k)$=0, $d(t_k)$=1.12 nm

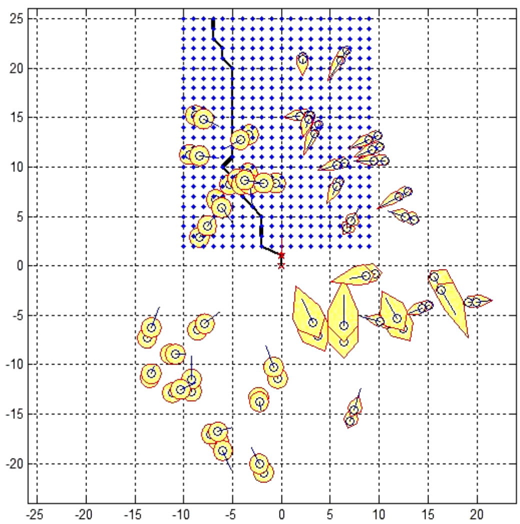

Figure 17. The safe trajectory of own ship for the DO algorithm in a situation of passing j=47 encountered ships, t_K^*=2.96 h

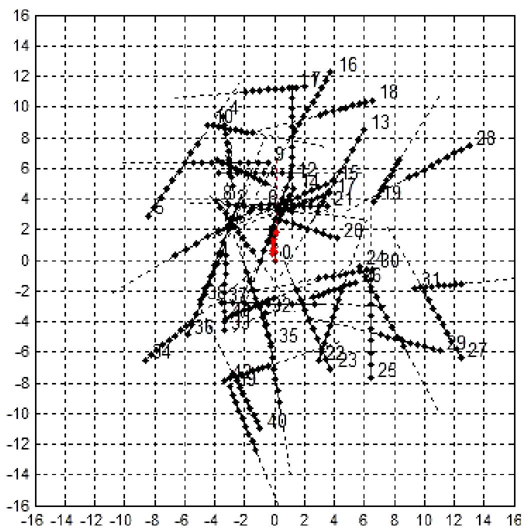

Figure 18. The safe trajectory of the own ship for the KO algorithm in a situation of passing j=47 encountered ships, $r(t_k)$=0, $d(t_k)$=0.19 nm

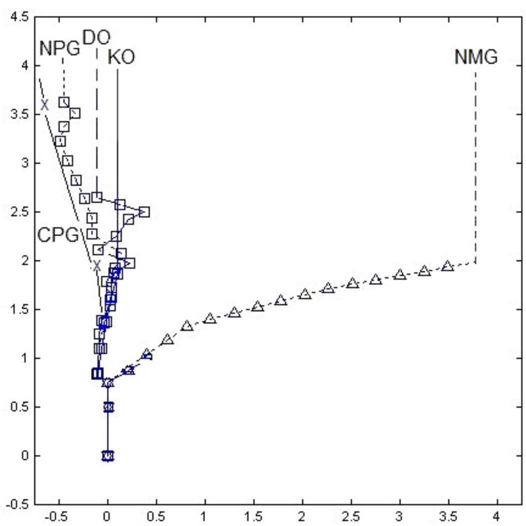

Figure 19. The comparison of safe trajectories of the own ship in the situation of j=47 -passing ships, designed with five algorithms: NPG, NMG, CPG, DO and KO

COLREG Rules. In DO and KO, the non-game algorithms, the changes of parameters of other ship movements are tracked by the anti-collision system ARPA and are taken into account in the algorithm of determining the safe trajectory of the own ship.

Comparison of simulation results of five algorithms presented in this paper and previously conducted research for a wide variety of real navigational situations at sea, confirm the conclusion that the DO algorithm is the most suitable one among those considered to constitute the computer assisted method for supporting the navigator in collision situations.

The DO algorithm is recommended for use in practice, as the subjectivity is taken into account, intervening when a navigator determines a safe manoeuver, this being in approximately 80% the cause of a collision of ships at sea.

References

BABA, N. and JAIN, L. C. (2001) *Computational Intelligence in Games.* Physica-Verlag, New York.

BASAR, T. and OLSDER, G. J. (1998) *Dynamic Noncooperative Game Theory.* SIAM, Philadelphia.

BIST, D. S. (2000) *Safety and Security at Sea.* Butter Heinemann, Oxford-New Delhi.

BOLE, A., DINELEY, B. and WALL, A. (2006) *Radar and ARPA Manual.* Elsevier, Amsterdam-Tokyo.

CAHILL, R. A. (2002) *Collisions and Their Causes.* The Nautical Institute, London.

COCKCROFT, A.N. and LAMEIJER, J. N. F. (2006) *The Collision Avoidance Rules.* Elsevier, Amsterdam-Tokyo.

CYMBAL, N. N., BURMAKA, I. A. and TUPIKOV, I. I. (2007) *Elastic strategies of passing the ships.* KP OGT, Odessa (in Russian).

ENGWERDA, J. C. (2005) *LQ Dynamic Optimization and Differential Games.* John Wiley and Sons, West Sussex.

GLUVER, H. and OLSEN, D. (1998) *Ship Collision Analysis.* A.A. Balkema, Rotterdam-Brookfield.

HASEGAWA, K., SHIGEMORI, Y. and ICHIYAMA, Y. (2000) Feasibility study on intelligent marine traffic system. In: *Proc. of the Maritime Conference of Manoeuvring Crafts*, Aalborg. IFAC, 121–132.

ISAACS, R. (1965) *Differential Games.* John Wiley & Sons, New York.

LEE, H. J. and RHEE, K. P. (2001) Development of collision avoidance system by using expert system and search algorithm. *International Shipbuilding Progress* **48**, 197-212.

LISOWSKI, J. (2013) The sensitivity of computer support game algorithms of a safe ship control. *International Journal of Applied Mathematics and Computer Science* **23** (2), 439-446.

MILLINGTON, I. and FUNGE, J. (2009) *Artificial Intelligence for Games.* Elsevier, Amsterdam-Tokyo.

MODARRES, M. (2006) *Risk Analysis in Engineering.* Taylor & Francis Group, Boca Raton.

NISAN, N., ROUGHGARDEN, T., TARDOS, E. and VAZIRANI, V. V. (2007) *Algorithmic Game Theory.* Cambridge University Press, New York.

OSBORNE, M. J. (2004) *An Introduction to Game Theory.* Oxford University Press, New York.

PIETRZYKOWSKI, Z. (2011) *The Navigational Decision Support System on a Sea-going Vessel.* Maritime University, Szczecin.

SEGHIR, M. M. (2012) Safe ship's control in a fuzzy environment using genetic algorithm. *Solid State Phenomena* **180**, 70-75.

SPEYER, J. L. and JACOBSON, D. H. (2010) *Primer on Optimal Control Theory.* SIAM, Philadelphia.

SZLAPCZYŃSKI, R. and ŚMIERZCHALSKI, R. (2009) Supporting navigators decisions by visualizing ship collision risk. *Polish Maritime Research* **59**, 83-88

ZIO, E. (2009) Computational methods for reliability and risk analysis. *Series on Quality, Reliability and Engineering Statistics* **14**, 295-334.