POLISH MARITIME RESEARCH 1(72) 2012 Vol 19; pp. 11-20
10.2478/v10012-012-0002-x

Evolutionary approach to ship's trajectory planning within Traffic Separation Schemes

Rafał Szłapczyński, Ph. D.
Gdańsk University of Technology

ABSTRACT

The paper presents the continuation of the author's research on evolutionary approach to ship trajectory planning. While the general problem of the evolutionary trajectory planning has already been solved, no one has yet touched one of its specific aspects: evolutionary trajectory planning within Traffic Separation Schemes. Traffic Separation Scheme (TSS) is a traffic-management route-system complying with rules of the International Maritime Organization. In brief, the ships navigating within a TSS all sail in the direction assigned to a particular traffic lane or they cross at a course angle as close to 90 degrees as possible. This and other TSS rules largely affect the evolutionary method. The paper presents a proposal of the extended evolutionary method, with a focus on changes that have to be made to obey TSS rules, especially the changes in the phases of evaluation and specialised operators of the evolutionary cycle.

Keywords: marine navigation; Traffic Separation Schemes; collision avoidance manoeuvres; evolutionary computing

INTRODUCTION

As for now, nearly all aspects of marine navigation have been subjects to scientific research, with most of the navigational processes having been modelled and the decision problems described formally. Especially, there are many numerical methods for planning ship routes or collision-avoidance manoeuvres. Even though some of them have not been applied yet, these problems are generally considered to be solved from the basic research point of view. Traffic Separation Scheme (TSS) rules however are an exception and have not been thoroughly investigated yet. In particular, there are no algorithms based on them while the rules themselves are too ambiguous to implement them directly.

The Traffic Separation Scheme (TSS) is a route management system complying with rules of the International Maritime Organization (IMO). TSS's are used to regulate the traffic at busy, confined waterways (usually port waters) or around capes. Typically a TSS consists of at least one traffic lane in each main-direction, turning-points, deep water lanes, separation zones between the main traffic lanes and inshore zones. The body of water between two opposite lanes is an area where navigating is only allowed under special circumstances (collision avoidance manoeuvres or crossing the lanes). As a result of lane separation, a risk for head-on collisions is greatly reduced. Where needed, there are special zones where a lane splits into two channels: one on-going and the other to the nearby ports. In most TTS schemes there are also Inshore Traffic Zones between the traffic-lanes and the coast. The inshore traffic zone is unregulated and should not be used for on-going traffic: it is meant for local traffic, fishing and small craft only.

There are a number of methods of solving multi-ship encounter situations: they can be divided into deterministic and heuristic ones. Deterministic approach is based on differential games and has been proposed by Lisowski [10]. Of the heuristic ones the most successful and flexible is searching for a ship's trajectory by genetic or evolutionary algorithms. The method has been first proposed by Smierzchalski and Michalewicz [12] and since then similar approaches has been tried by other researchers: evolutionary computation (EC) may be applied for finding an optimal path [16, 21] and genetic algorithms (GA) are used for optimization of collision avoidance manoeuvres [8, 18]. Other related approaches include trajectory optimization by using genetic annealing algorithm [1] and ship collision avoidance route planning by ant colony algorithm [17]. Summaries of applying GA and EC to maritime collision avoidance and trajectory planning have been presented in [13, 20] among others. In short, EC and GA approach to the problem use algorithms which for a given set of pre-determined input trajectories find a solution that is optimum according to a given fitness function. In this paper is introduced an extension to the evolutionary approach previously proposed by the author [15], which allows the evolutionary method to support special rules that have to be obeyed within Traffic Separation Schemes.

Handling the TSS rules affects two phases of the evolutionary method: evaluation (where breaking the rules is detected and penalized by fitness function) and specialised operators which aim at eliminating the violations of the rules by adjusting the trajectories of ships. Therefore it is these two phases that the paper focuses on. The rest of the paper is organized as follows. In the next section a more formal description of the problem is given, including the particular TSS rules which have to be obeyed. Then the ways of detecting constraints violations, including violations of TSS rules, are discussed. This is followed by the summarised foundations of the evolutionary method, with a brief explanation of each of the phases of the cycle. Then detailed descriptions of phases of interest (evaluation and specialised operators) are provided. Finally example results of the method and the conclusions are presented.

OPTIMISATION PROBLEM

It is assumed that the following data are given:
- stationary constraints (such as landmasses and other obstacles),
- positions, courses and speeds of all ships involved,
- ship domains,
- times necessary for accepting and executing the proposed manoeuvres.

Ship positions and ship motion parameters are provided by ARPA (Automatic Radar Plotting Aid) and AIS (Automatic Identification System) systems. A ship domain can be determined on the basis of the ship's length, its motion parameters and type of water region. Since the shape of a domain is dependent on type of water region, the author has decided to use a ship domain model by Davis [5], which updated Goodwin model [7], for open waters and to use a ship domain model by Coldwell [3], which updated Fuji model [6], for restricted waters. The last parameter – the necessary time, is computed on the basis of navigational decision time and the ship's manoeuvring abilities. By default a 6-minute value is used here.

There are also some additional COLREGS-related assumptions, namely:
- the method is applied for good visibility conditions,
- all considered ships are equally privileged,
- all considered ships have motor engines and are not restricted in their ability to manoeuvre: no sailing ships or damaged ships are taken into account – such ships should be always given way to (COLREGS, Rule18 - responsibilities between vessels),

Knowing all the above mentioned parameters, the goal is to find a set of trajectories which minimizes the average way loss spent on manoeuvring, while fulfilling the following conditions:
- none of the stationary constraints are violated,
- none of the ship domains are violated,
- the minimal acceptable course alteration should not be too small (the minimal alteration of 15 degrees has been assumed here),
- the maximal acceptable course alteration should not be too large (the maximal alteration of 60 degrees has been assumed here),
- speed alteration are not to be applied unless necessary (collision cannot be avoided by course alteration up to 60 degrees),
- a ship only manoeuvres when she is obliged to,
- in case of head-on and crossing encounters, manoeuvres to starboard are favoured over manoeuvres to port board.

As for the first two conditions, all obstacles have to be avoided and the ship domain is an area that should not be violated by definition. All the other conditions are either imposed by COLREGS [2, 4] and good marine practice or by economics. In particular, very small course alterations might be misleading for the ARPA systems (and therefore may lead to collisions) and very large course alterations are not recommended due to efficiency reasons. Also, ships should only manoeuvre when necessary, since each manoeuvre of a ship makes it harder to track its motion parameters for the other ships' ARPA systems. Apart from the main constraints, additional constraints – selected COLREGS rules have to be directly handled. In brief, the basic COLREGS rules of interest here are:
- Rule 13 – overtaking: an overtaking vessel must keep well clear of the vessel being overtaken.
- Rule 14 - head-on situations: when two power-driven vessels are meeting head-on both must alter course to starboard so that they pass on the port side of the other.
- Rule 15 - crossing situations: when two power-driven vessels are crossing, the vessel, which has the other on the starboard side must give way.
- Rule 16 - the give-way vessel: the give-way vessel must take early a substantial action to keep well clear.
- Rule 17 - the stand-on vessel: the stand-on vessel may take action to avoid collision if it becomes clear that the give-way vessel is not taking appropriate action.

The behaviour of ships within TSS is specified by Rule 10 whose key points are as follows:

a) A vessel using a traffic separation scheme shall:
   - proceed in the appropriate traffic lane in the general direction of traffic flow for that lane,
   - so far as practicable keep clear of a traffic separation line or separation zone,
   - join or leave a traffic lane at the termination of the lane, but when joining or leaving from either side shall do so at as small an angle to the general direction of traffic flow as practicable.

b) A vessel shall so far as practicable avoid crossing traffic lanes, but if obliged to do so shall cross on a heading as nearly as practicable at right angle to the general direction of traffic flow.

c) A vessel shall not use an inshore traffic zone when she can safely use the appropriate traffic lane within the adjacent traffic separation scheme unless:
   - it is a vessel of less than 20 metres in length, sailing vessel or that engaged in fishing,
   - it is on route to or from a port, offshore installation or structure, pilot station or any other place situated within the inshore traffic zone, or to avoid immediate danger.

d) A vessel other than a crossing vessel or a vessel joining or leaving a lane shall not normally enter a separation zone or cross a separation line except:
   - in cases of emergency to avoid immediate danger,
   - to engage in fishing within a separation zone.

In the following sections it is analysed how the violations of all these constraints can be detected, and how severely should they be penalized during the evaluation phase by the fitness function of the evolutionary method.

DETECTING VIOLATIONS OF CONSTRAINTS

Detecting violations of static constraints (collisions with landmasses and safety isobaths)

The method uses a vector map of a given area. However it is not processed directly for detection of constraints violations, but for generating bitmap of an area, which is done offline and only once for each area. Then, when the method is running in real time, each bitmap cell, which the trajectory of a ship traverses, is read and checked for belonging to landmass or safety isobaths. For a bitmap whose detail level reflects this of a given vector map, the computational time would be proportional to the number of traversed cells and thus acceptable. This approach is also quite flexible in terms of future implementation of bathymetry: if every cell contained information on the water depth, it would be easy to check whether a cell is passable or not for a particular ship.

Detecting ship domain violations

The algorithm for detecting ship-to-ship collisions is as follows. Each ship's trajectory is checked against all other ships. For each pair of ships, the start time and end time of each trajectory's segments are computed. If two segments of the two trajectories overlap in time, they are checked for geometrical crossing. In case of a crossing, the approach factor value is computed. Then, if the approach factor value indicates collision, the type of an encounter (head-on, crossing or overtaking) is determined on the basis of the ships' courses and it is decided, which ship is to give way (both ships in case of head-on). The collision is only registered for the give-way ship and the information on the collision are stored in the trajectory's data structure.

Detecting general COLREGS violations

The three most common types of COLREGS violations are as follows:
- a ship does not give way, when it should,
- a ship gives way, when it should not (making unexpected and misleading manoeuvres),
- a ship manoeuvres to port-board when it should manoeuvre to starboard.

Each of these three situations may happen on either open or restricted waters, which gives a total of six cases to handle. Unfortunately, when analysing a set of ship trajectories for a multi-target encounter, it is sometimes impossible to recognize the reason for a particular manoeuvre: which ship was given way intentionally, and which one benefited from it only as a side effect. As a result, not all violations can be detected. The final COLREGS violations detection rules applied in the method are:

1) In open waters:
   a) if a ship is not obliged to give way to any other ship, any manoeuvre it performs is registered as COLREGS violation,
   b) if a ship is obliged to give way and does not perform a manoeuvre, it is registered as COLREGS violation,
   c) all manoeuvres to port board are registered as COLREGS violations.
2) In restricted waters: here, as explained before, every trajectory node which is a part of a manoeuvre, contains information on the reason why this particular node has been inserted or shifted: land or other stationary obstacle avoidance, target avoidance or accidental manoeuvre generated by evolutionary mechanisms. Based on this, COLREGS violations are registered as follows:
   a) if a ship does not initially have to give way to any target and its first manoeuvre has reason other than static constraint violation avoidance, it is registered as COLREGS violation,
   b) any manoeuvre to port board of reason other than static constraint violation avoidance is registered as COLREGS violation.

Detecting TSS rules violations

The possible lawful types of routes which are the results of applying the COLREGS rule 10 are shown in Fig. 1. For a TSS sector shown in Fig. 1 the preferred tracks that the ships should follow are:
- Track A – through traffic,
- Track B – traffic using a lane and crossing another lane to reach inshore zone; notice the small angle at which it leaves the lane to reach separation zone and altering course within the separation zone,
- Track C – traffic crossing TSS at right angle,
- Track D – traffic joining lane from the side,
- Track E – traffic leaving the inshore zone, crossing one lane and joining the other lane,
- Track F – traffic leaving the lane at small angle.

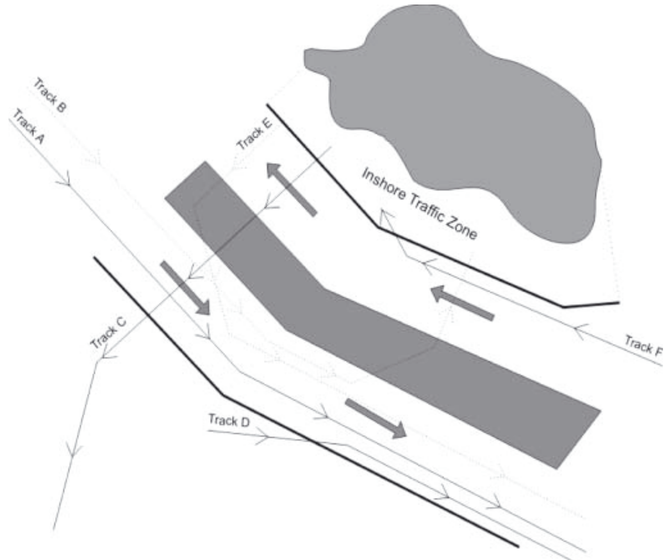

Fig. 1. *TSS and different routes through a sector*

The algorithm for TSS rules' violation detection is given as follows. The first step for checking for a TSS rule violation is always the checking if a trajectory's segment has crossed with a certain TSS system. If so, then each of the TSS's parts will be checked. Since each part of a TSS is a polygon it is enough to detect a crossing of a trajectory's segment with one of the polygon's edges. Once such crossing has been detected the further conditions for violations will be checked, depending on the particular TSS part. The following types of TSS violations have been identified and used in this research (the violations have been grouped into three major categories, which are shown in Fig. 2-4):

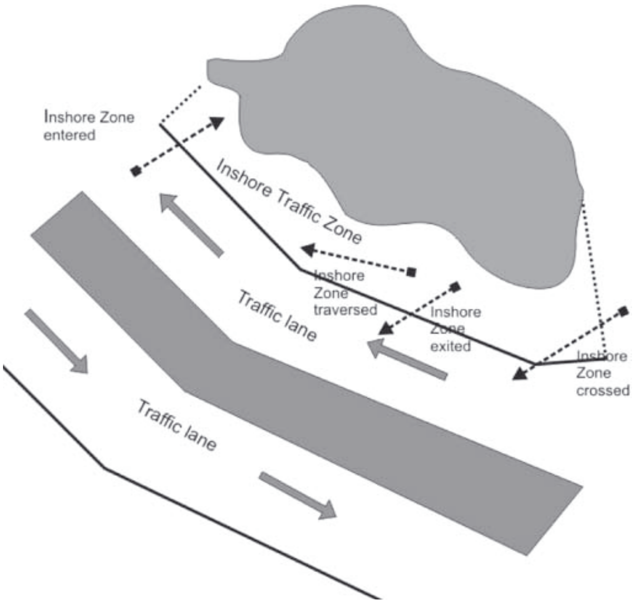

Fig. 2. Violations of Inshore Zone

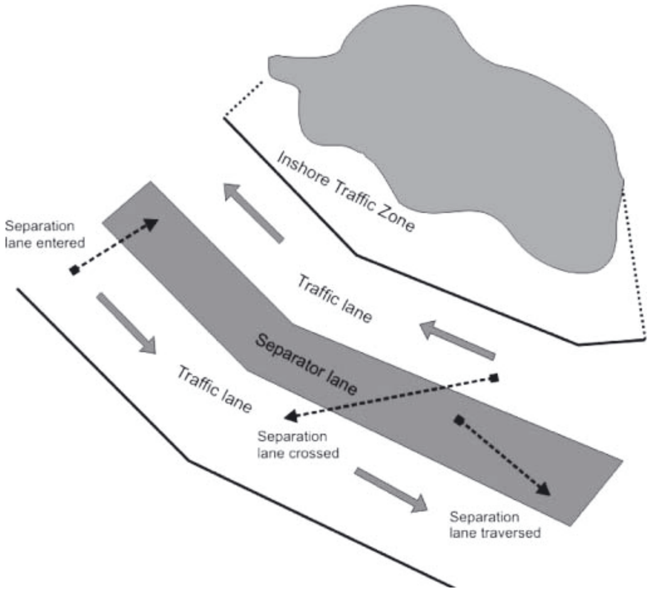

Fig. 3. Violations of Separation lane

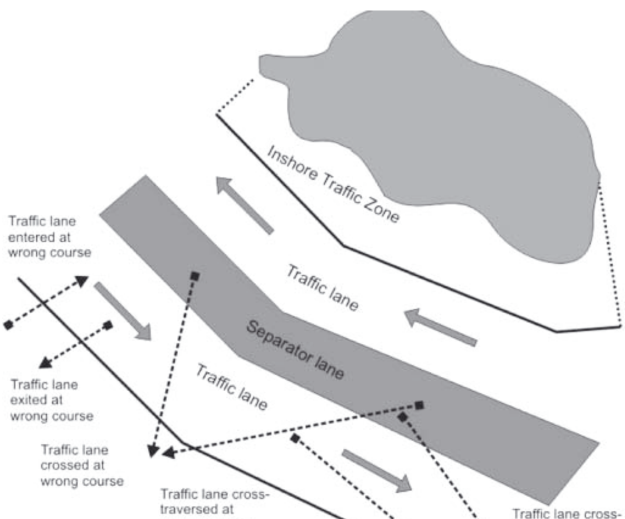

Fig. 4. Violations of traffic lane

1. Inshore Zone entered: the first endpoint of a trajectory's segment is outside the Inshore Zone and the second one is inside.
2. Inshore Zone traversed: both endpoints of a trajectory's segment are inside the Inshore Zone.
3. Inshore Zone crossed: both endpoints of a trajectory's segment are outside the Inshore Zone and the segment crosses Inshore Zone boundaries.
4. Inshore Zone exited: the first endpoint of a trajectory's segment is inside the Inshore Zone and the second one is outside.
5. Traffic lane entered at a wrong course: the first endpoint of a trajectory's segment is outside the traffic lane and the second one is inside. The course of entrance differs from the lane's direction by more than 20 degrees.
6. Traffic lane traversed at a wrong course: both endpoints of a trajectory's segment are inside the lane. The course of a segment differs from the lane's direction by more than 5 degrees.
7. Traffic lane cross-traversed at a wrong course: both endpoints of a trajectory's segment are outside the lane. The course of a segment differs from the lane's direction by more than 5 degrees but less than 45 degrees.
8. Traffic lane cross-traversed at a wrong direction: both endpoints of a trajectory's segment are outside the lane. The course of a segment differs from the lane's direction by more than 135 degrees.
9. Traffic lane crossed at a wrong course: both endpoints of a trajectory's segment are outside the lane. The course of a segment differs from the perpendicular to the lane's direction by more than 5 degrees and less than 45 degrees.
10. Traffic lane exited at a wrong course: the first endpoint of a trajectory's segment is inside the traffic lane and the second one is inside? The course of a segment differs from the lane's direction by more than 20 degrees.
11. Separation lane entered: the first endpoint of a trajectory's segment is outside the separation lane and the second one is inside.
12. Separation lane traversed: both endpoints of a trajectory's segment are inside the separation lane.
13. Separation lane crossed: both endpoints of a trajectory's segment are outside the separation lane.
14. Separation lane exited: the first endpoint of a trajectory's segment is inside the separation lane and the second one is outside.

The identified violation types of TSS rules are shown in Fig. 2 (Inshore Zone violations), Fig. 3 (Separation lane violations) and Fig. 4 (Traffic lane violations).

EVOLUTIONARY COMPUTING - GENERAL IDEA

Before proceeding further with the details of the method which solves the above formulated optimization problem, some basic information on evolutionary programming are provided in this section. The general idea of evolutionary computing [11] is shown in Fig. 5.

First, the initial population of individuals (each being a potential solution to the problem) is generated either randomly or by other methods. Usually none of the individuals is optimal or even close to that. Sometimes none of the individuals is acceptable. This initial population is a subject of subsequent iterations of evolutionary algorithm. Each of the iterations consists of the following steps:

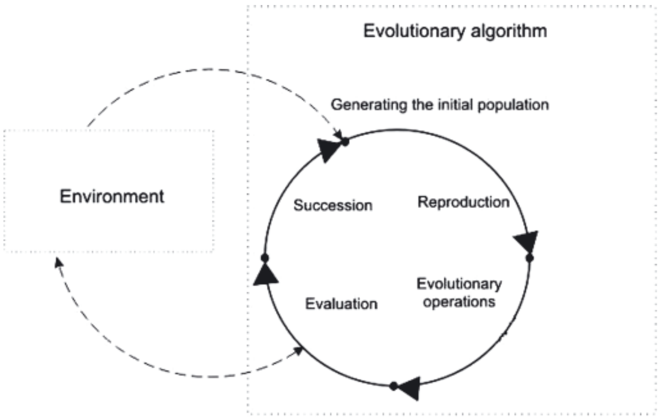

Fig. 5. Evolutionary algorithms - general idea

1. Reproduction: sets of parents (usually pairs) are selected from all of the individuals and they are crossed to produce offspring. The offspring inherits some features from each parent.
2. Evolutionary operations: the offspring is modified by means of random mutation operators as well as specialized operators dedicated to the problem.
3. Evaluation: each of the individuals (including parents and the offspring) is assigned a value of a fitness function which reflects the quality of the solution represented by this individual.
4. Succession: the next generation of individuals is selected. The selection is based on the results of the evaluation. Usually the individuals are chosen randomly with the probability strictly depending on the fitness function value.

The evolutionary algorithm ends when one of the following happens:

- maximum acceptable time or number of iterations is reached,
- the satisfactorily high value of fitness function has been reached by one of the individuals,
- further evolution brings no improvement.

The main difference between the evolutionary computing and pure genetic algorithms is that in the former the individuals directly represent the potential solutions to the problem, without being translated to chromosomes first. This allows for specialised operators dedicated to the problem, which, for some classes of optimisation tasks, greatly speed up the evolutionary process, resulting in a much lesser number of generations needed and much lower computational time. The method discussed in the paper, focuses on specialised operators thus fully utilizing the possibilities of evolutionary computing.

EVALUATION

In the evolutionary method all individuals (sets of trajectories) are evaluated by the specially designed fitness function which should reflect optimisation criteria and constraints [11]. In this section it is shown how this fitness function is formulated.

Basic criterion – minimizing way loss

The basic criterion is the economic one – minimizing way loss of a ship. A trajectory_economy_factor is computed according to the formula (1).

$$\text{trajectory\_economy\_factor} = \left(\frac{\text{trajectory\_length} - \text{way\_loss}}{\text{trajectory\_length}}\right) \tag{1}$$

where:

trajectory_length – the total length of the ship's trajectory [nautical miles],

way_loss – the total way loss of the ship's trajectory [nautical miles] computed as a difference between the trajectory length and length of a segment joining trajectory's start point and endpoint.

As can be seen, the trajectory_economy_factor is always a number from a (0,1] range.

Penalizing static constraint violation

After the trajectory economy factor has been computed the static constraints are handled by introducing penalties for violating them. For each trajectory its static constraint factor scf is computed. The static constraints are always valid and their violations must be avoided at all costs, therefore penalties applied here are the most severe – hence the square in the formula (2).

$$\text{scf} = \left(\frac{\text{trajectory\_length} - \text{trajectory\_cross\_length}}{\text{trajectory\_length}}\right)^2 \tag{2}$$

where:

trajectory_cross_length – the total length of the parts of the ship's trajectory which violate stationary constraints [nautical miles].

The static constraint factor is a number from a [0,1] range, where "1" value means no static constraint violation (no landmasses or other obstacles are crossed) and "0" value is for trajectories crossing landmasses on their whole length.

Penalizing collisions with other ships

Analogically to the static constraint factor, collision avoidance factor caf is computed to reflect the ship's collisions with all other privileged ships as shown by (3).

$$\text{caf} = \prod_{j=1}^{n} [\min(\text{fmin}_j, 1)]^2 \tag{3}$$

where:

n – the number of ships [/],

j – the index of a target ship [/],

f_{min_j} – the approach factor [14] value for an encounter with ship j, if the own ship is the privileged one, the potential collision is ignored and the approach factor value is equal to 1 by definition. [/].

The collision avoidance factor is a number from a [0,1] range, where "1" value means no ship domain violation and "0" means a crash with at least one of the targets.

Penalizing TSS rules violations

TSS compliance factor tcf is computed according to the following formula (4).

$$\text{tcf} = 1 - \sum_{k=1}^{m} [\text{TSS\_violation\_penalty}_k] \tag{4}$$

where:

m – he number of TSS rules violation registered for the current ship [/],

k – the index of a registered violation [/],

$\text{TSS\_violation\_penalty}_k$ – the penalty for the k-th of the registered TSS rules violation [/].

The TSS rules violations are penalized as follows:

1. through 4. (Inshore Zone violations):

$$\text{TSS\_violation\_penalty} = \\ = 2 \cdot \text{segment\_violation\_percentage} \quad (5)$$

5. and 10. (Traffic Lane is entered or exited at a wrong course):

$$\text{TSS\_violation\_penalty} = \\ = 2 \cdot \text{segment\_violation\_percentage} \cdot \\ \cdot \sin(\text{e\_dev\_angle}) \quad (6)$$

where:

$$\text{e\_dev\_angle} = |\text{traffic\_lane\_course} - \text{ship\_course}| - 20 \quad (7)$$

6. and 7. (Traffic Lane is traversed at a wrong course or Traffic lane is cross-traversed at a wrong course):

$$\text{TSS\_violation\_penalty} = \\ = 2 \cdot \text{segment\_violation\_percentage} \cdot \\ \cdot \sin(\text{t\_dev\_angle}) \quad (8)$$

where:

$$\text{t\_dev\_angle} = |\text{traffic\_lane\_course} - \text{ship\_course}| - 5 \quad (9)$$

8. (Traffic lane is cross-traversed at a wrong direction):

$$\text{TSS\_violation\_penalty} = \quad (10) \\ = 2 \cdot \text{segment\_violation\_percentage} \cdot \sin\left(\frac{\text{ct\_dev\_angle}}{2}\right)$$

where:

$$\text{ct\_dev\_angle} = |\text{traffic\_lane\_course} - \text{ship\_course}| - 5 \quad (11)$$

9. (Traffic lane is crossed at a wrong course):

$$\text{TSS\_violation\_penalty} = \\ = 2 \cdot \text{segment\_violation\_percentage} \cdot \\ \cdot \sin(\text{c\_dev\_angle}) \quad (12)$$

where:

$$\text{c\_dev\_angle} = |\text{perpendicular\_to\_ traffic\_lane\_course} - \\ + \text{ship\_course}| - 5 \quad (13)$$

11. through 14. (Separation Lane violations):

$$\text{TSS\_violation\_penalty} = \\ = \frac{\text{segment\_violation\_percentage}}{2} \quad (14)$$

For all penalty formulas:

$$\text{segment\_violation\_percentage} = \\ = \frac{\text{violating\_part\_length}}{\text{segment\_length}} \quad (15)$$

Penalizing other COLREGS violations

Within TSS, the other COLREGS violations are secondary to static constraint violations, collisions with other ships and TSS rules violation and therefore the author has decided to penalize it moderately to make sure that constraints from the previous two points are met first. COLREGS compliance factor ccf is computed according to the following formula:

$$ccf = 1 - \sum_{k=1}^{m} [\text{COLREGS\_violation\_penalty}_k] \quad (16)$$

where:

m – the number of COLREGS violation registered for the current ship [/],

k – the index of a registered violation [/],

$\text{COLREGS\_violation\_penalty}_k$ – the penalty for the k-th of the registered COLREGS violation [/].

The penalty values for all registered COLREGS violations are configurable in the method and are set to 0.05 by default.

Fitness function value

Once all aforementioned factors have been computed the normalized fitness function value is calculated. The fitness function is normalized for convenience of further evolutionary operations, mostly for selection purposes. As a result no additional operations on fitness function values are necessary and they can directly be used for random proportional and modified random proportional selection in the reproduction and succession phases of the evolutionary algorithm. It is also easier to measure and see progress that is made with each generation. However, normalized fitness function is harder to obtain, because one has to make sure that the high resolution of evaluating the individuals is kept, namely that various levels of penalties are used: stationary constraints, being more important than collision avoidance and collision avoidance being more important than COLREGS compliance. Here, the normalized fitness function keeps relatively high resolution of evaluation: minor stationary constraints violations are penalized similarly as major collisions with other ships and minor collisions with other ships are penalized similarly as multiple COLREGS violations. The final fitness function is as follows:

$$\text{trajectory\_fitness} = \\ = \text{trajectory\_economy\_factor} \cdot scf \cdot caf \cdot tcf \cdot ccf \quad (17)$$

SPECIALISED OPERATORS WHICH FIX TSS RULES' VIOLATIONS

The random mutation operators and specialised operators dedicated for static constraints' violations problem and ship collisions' problem have been introduced in [15]. Therefore in this section only new, previously not presented operators which fix TSS rules' violations are described here. In the method's data structure all TSS violations are stored and their data contains:

- VTS violation type
- numbers which identify a TSS and its part which has been violated

- a segment responsible for the violation (identified by its first node)
- recommended course for this TSS part
- violation angle (the difference between recommended course and the segment's course)
- percentage of a segment's length which violates the rule
- coordinates of entrance point (if a segment enters a TSS part)
- coordinates of exit point (if a segment exits a TSS part)

The designed operators are semi-deterministic [9], that is based on these data, and it is decided what action can be taken to fix the part of a trajectory which violates the TSS rules. The actions can be roughly classified as:
- avoiding entering a certain TSS part,
- adjusting the current course to the recommended course.

Below the operators used for fixing violations of inshore zone, separator lanes and traffic lanes are briefly described.

Fixing violations of inshore zone

For violations of inshore zone the action is always avoiding entering this part of a TSS. The operators which are used for this are:
- inserting a new segment,
- inserting a node,
- shifting the first node of a segment,
- shifting the second node of a segment,
- shifting a segment.

These operators work very similarly as collision voidance operators described in [15]. A particular operator is chosen, based on the time distance between two nodes, which constitute the violating segment and on the time remaining to violation when the ship is in the first node. It is assumed that a navigator should always have a predefined amount of time for decision making and manoeuvre execution (by default – 6 minutes) and therefore the time between two subsequent manoeuvres cannot be shorter than this predefined time. Inserting a new segment (two course alteration manoeuvres) or a new node (a single course alteration manoeuvre) should not affect the minimal time space between manoeuvres. Therefore, the choice of an operator is done as follows:
- whenever there is enough time, a new segment is inserted
- if a new segment cannot be inserted, a new node is inserted, if possible
- if a new node cannot be inserted, the violating segment or one of its nodes is moved away from the violation point (usually the entrance point). Usually the node closer to the violation point is moved or a segment is moved if the violation point is close to the middle of this segment.

The operators are illustrated in Fig. 6 through 10.

Fixing violations of separator lanes

It has been decided that violations of separator lanes, which are penalized moderately, will not be fixed because they are often side effects of collision avoidance manoeuvres. The only exception is crossing a separator lane at a wrong course. In such cases, if possible, a new node is inserted so as to cross at a course perpendicular to the lane direction. The operator is illustrated in Fig. 11.

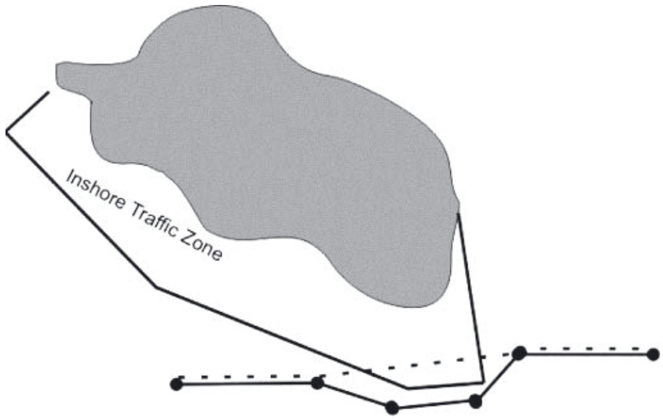

Fig. 6. *Fixing violations of inshore zone - inserting a new segment*

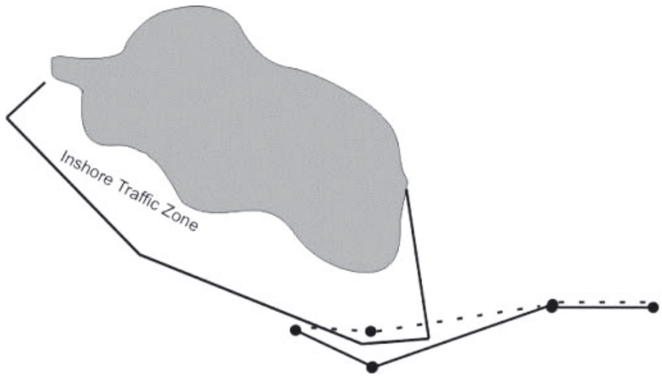

Fig. 7. *Fixing violations of inshore zone - inserting a new node*

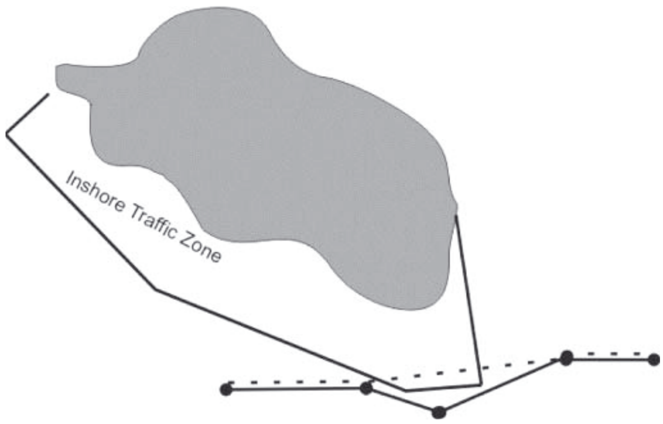

Fig. 8. *Fixing violations of inshore zone -shifting the first node of a segment*

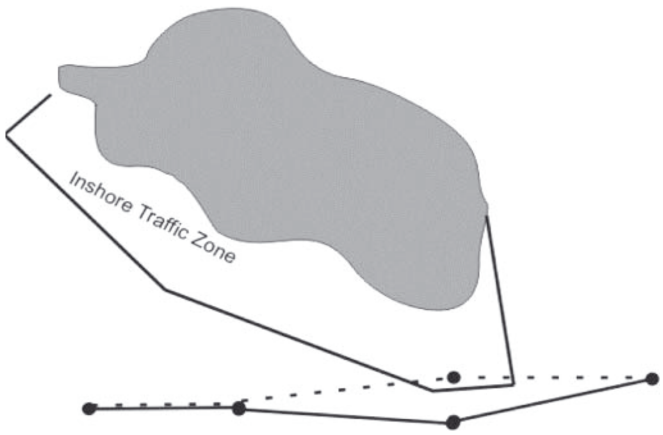

Fig. 9. *Fixing violations of inshore zone - shifting the second node of a segment*

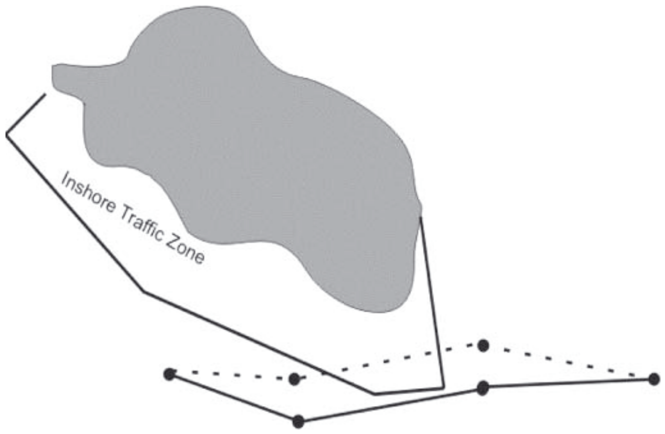

Fig. 10. Fixing violations of inshore zone - shifting a segment

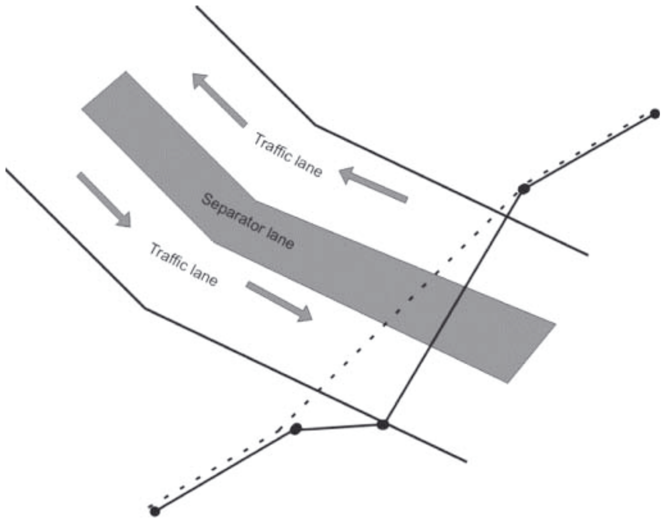

Fig. 11. Fixing violations of separator lane - crossing the separator lane

Fixing violations of traffic lanes

The fixing operators here work as follows:

- in case of entering or exiting at a wrong course, the second node of the segment is moved so as to change the current course to the one differing by 20 degrees from the lane direction
- in case of traffic lane traversed at a wrong course (differing from the recommended one by more than 5 degrees), the second node is moved so as to adjust the course to the recommended one. If this is not possible due to the course difference being too large, the second node of the segment is deleted.
- in case of a traffic lane crossed at wrong course, a new node is inserted so as to cross at a course perpendicular to the lane's direction, similarly as in case of crossing a separator lane
- in case of a traffic lane cross-traversed at a wrong direction, the violation is fixed similarly as in case of inshore zone violation, that is, an avoidance action is chosen
- in case of a traffic lane cross-traversed at wrong course, a new segment is inserted between the nodes of the violating segment so as to adjust movement within the lane to the traffic lane direction. If such action generates entering or exiting at wrong course, the second node of the violating segment will be additionally moved so as to change the current course to the one differing by 20 degrees from the lane direction.

The operators are illustrated in Fig. 12 through 15.

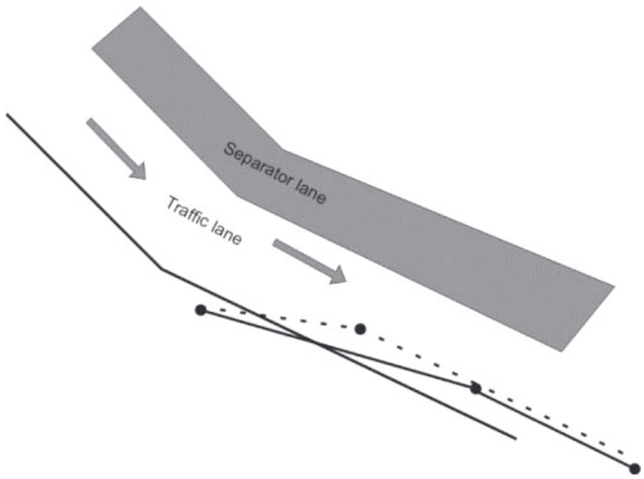

Fig. 12. Fixing violations of traffic lane - entering or exiting the lane

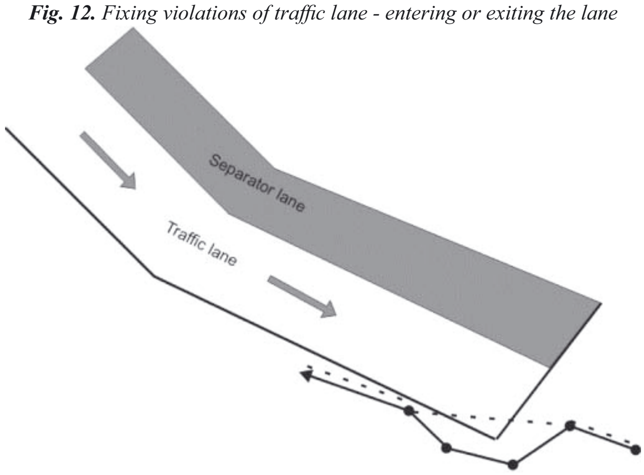

Fig. 13. Fixing violations of traffic lane - traversing the lane

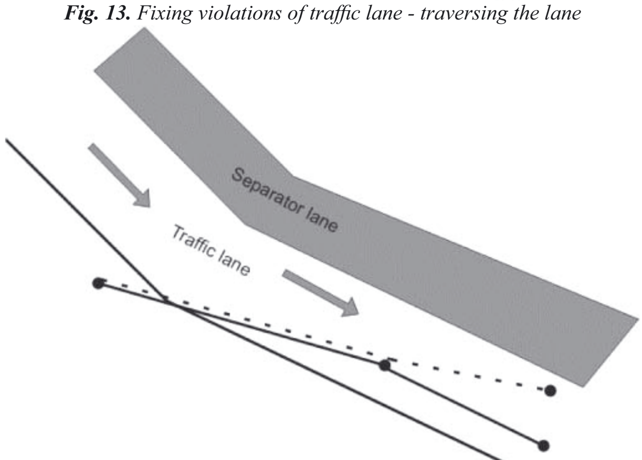

Fig. 14. Fixing violations of traffic lane - avoiding the movement in wrong direction

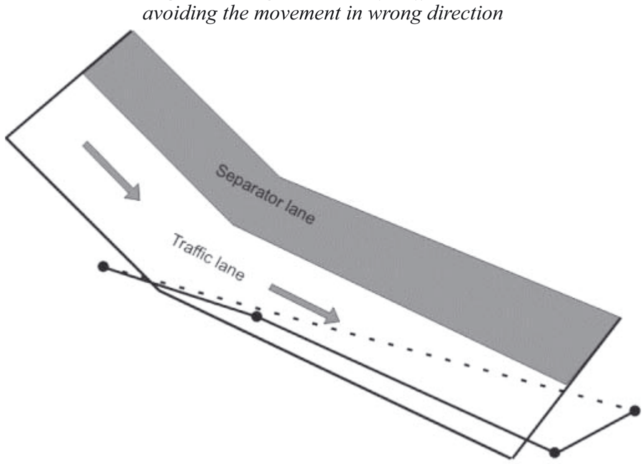

Fig. 15. Fixing violations of traffic lane - cross-traversing the lane

EXAMPLES OF RESULTS OF SIMULATIONS

In this section examples of ship routes planned by the method are shown (Fig. 16 and 17). Both scenarios are set in the Traffic Separation Scheme "Gulf of Gdansk" which consists of the following elements:

a) "TSS-WEST" (incoming lane and outgoing lane separated by a line – these traffic lanes are shown in a left part of each figure)
b) "TSS-EAST" (incoming lane and outgoing lane separated by a separator lane – these traffic lanes are shown in a right part of each figure)
c) "ITZ" - Inshore Traffic Zones (shown in a left part of each figure and marked with a red dotted line)
d) Recommended tracks (additional routes).

In Fig. 16 a ship traverses the incoming traffic lane of "TSS-WEST" and avoids violations of Inshore Traffic Zone, separator line and outgoing traffic lane. The ship's course is parallel to the recommended direction throughout the lane's length.

In Fig. 17 it is shown how a ship changes its course to avoid violating "TSS-WEST" and then crosses two traffic lanes and a separator lane of "TSS-EAST". The crossing is done perpendicularly to the lanes' direction.

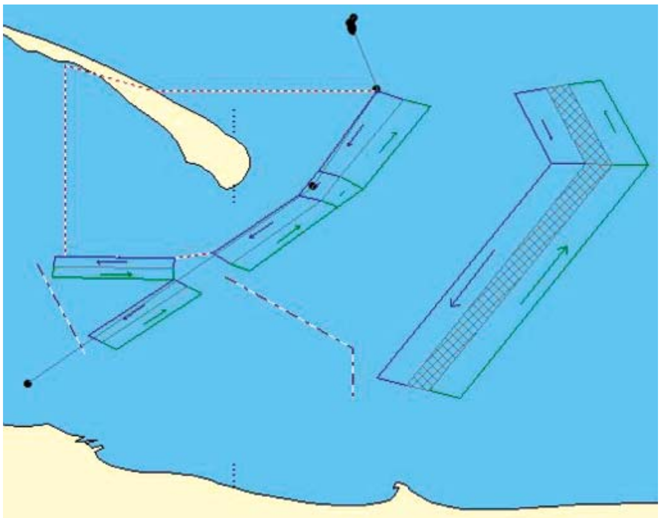

Fig. 16. *A ship traversing an incoming traffic lane*

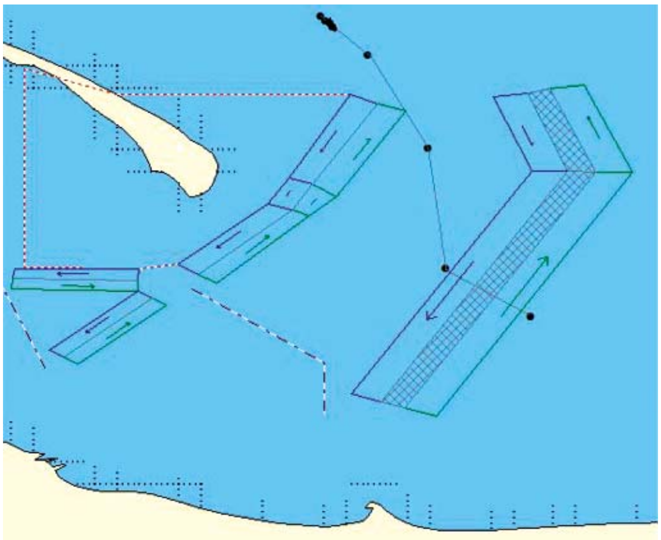

Fig. 17. *A ship crossing traffic lanes and a separator laneIn both cases the ships' behaviour is compliant with TSS recommendations specified by COLREGS Rule 10*

SUMMARY AND CONCLUSIONS

In this paper an evolutionary trajectory planning method has been presented. The description has been focused on supporting rules abiding within Traffic Separation Schemes, which have not been handled before by similar evolutionary methods developed by other researchers. In the course of this work the following tasks have been solved. First, TSS violations have been grouped into categories. Then based on these categories, fitness function including penalties for TSS violations has been proposed. Finally a set of specialized TSS-dedicated operators has been designed to increase the effectiveness of fixing TSS violations. The preliminary tests of the method have been carried out and they have confirmed the validity of the presented approach. Examples of the method's results has been provided to illustrate how it finds a satisfying solution for basic situations. Further research of the author will be concentrated on a generalization of the presented method. The envisaged future version of the method would return a set of trajectories of all ships within jurisdiction of a VTS system supervising a given TSS region.

BIBLIOGRAPHY

1. Cheng X., Liu Z.: Trajectory Optimization for Ship Navigation Safety Using Genetic Annealing Algorithm. ICNC 2007. Third International Conference on Natural Computation, 2007, vol. 4, pp. 385-392.
2. Cockroft A.N., Lameijer, J.N.F.: A Guide to Collision Avoidance Rules. Butterworth-Heinemann Ltd.,1993.
3. Coldwell T.G.: Marine Traffic Behaviour in Restricted Waters. The Journal of Navigation, 1983, 36, 431-444.
4. COLREGS: Convention on the International Regulations for Preventing Collisions at Sea. 1972.
5. Davis P.V., Dove M.J., Stockel C.T.: A Computer Simulation of multi-Ship Encounters. The Journal of Navigation, 1982, 35, 347-352.
6. Fuji J., Tanaka K.: Traffic Capacity. The Journal of Navigation,1971, 24, 543-552.
7. Goodwin E.M.: A Statistical Study of Ship Domains. The Journal of Navigation, 1975, 28, 329-341.
8. Ito M., Feifei Z., Yoshida N.: Collision avoidance control of ship with genetic algorithm. Proceedings of the 1999 IEEE International Conference on Control Applications, 1999, vol. 2, 1791 - 1796.
9. Jozwiak L., Postula, A.: Genetic engineering versus natural evolution: genetic algorithms with deterministic operators. Journal of Systems Architecture, 2002, 48, 99-112.
10. Lisowski J.: The Dynamic Game Models of Safe Navigation. International Journal on Marine Navigation and Safety of Sea Transportation, 2007, Vol. 1, No. 1.
11. Michalewicz Z., Fogel D.B.: How To Solve It: Modern Heuristics, Springer - Verlag, 2004.
12. Smierzchalski R., Michalewicz Z.: Modeling of a Ship Trajectory in Collision Situations at Sea by Evolutionary Algorithm. IEEE Transactions on Evolutionary Computation, 2000, No. 3, Vol. 4, 227-241.
13. Statheros T., Howells G., McDonald-Maier K.: Autonomous Ship Collision Avoidance Navigation Concepts, Technologies and Techniques.The Journal of Navigation, 2008, 61, 129–142.
14. Szlapczynski R.: A unified measure of collision risk derived from the concept of a ship domain. The Journal of Navigation, 2006, 59, 477-490.
15. Szlapczynski R.: Evolutionary Sets Of Safe Ship Trajectories: A New Approach To Collision Avoidance. The Journal of Navigation, 2011, 64, 169-181.
16. Tam C.K., Bucknall R.: Path-planning algorithm for ships in close-range encounters. Journal of Marine Science Technology, 2010, vol. 15, 395–407.

17. Tsou M. C., Hsueh C. K.: The study of ship collision avoidance route planning by ant colony algorithm. Journal of Marine Science and Technology, 2010, 18(5), 746–756.
18. Tsou M.-C., Kao S.-L., Su C.-M.: Decision Support from Genetic Algorithms for Ship Collision Avoidance Route Planning and Alerts. The Journal of Navigation, 2010, 63, 167-182.
19. Xue Y., Lee B.S., Han D.: Automatic collision avoidance of ships. Proceedings of the Institution of Mechanical Engineers, Part M: Journal of Engineering for the Maritime Environment, 2009, 33-46.
20. Yang L.L., Cao S.- H.. Li B.Z.: A Summary of Studies on the Automation of Ship Collision Avoidance Intelligence. Journal of Jimei University (Natural Science), 2006, vol. 2.
21. Zeng X.: Evolution of the safe path for ship navigation. Applied Artificial Intelligence, 2003, 17, 87–104.

CONTACT WITH AUTHOR

Rafał Szłapczyński, Ph. D.
Faculty of Ocean Engineering
and Ship Technology,
Gdańsk University of Technology
Narutowicza 11/12
80-233 Gdańsk, POLAND
e-mail: rafal@pg.gda.pl