

Adaptive Modeling of a Ship Trajectory in Collision Situations at Sea

Roman Smierzchalski, Gdynia Maritime Academy, Gdynia, Poland, *roms@wsm.gdynia.pl*
Zbigniew Michalewicz, University of North Carolina, Charlotte, USA *and*
Polish Academy of Sciences, Warsaw, Poland, *zbyszek@uncc.edu*

Abstract

For given circumstances (i.e., a collision situation at sea), a decision support system should help the operator to choose a proper maneuver, teach him/her good habits, and enhance his/her general intuition on how to behave in similar situations in the future. By taking into account certain boundaries of the maneuvering region, along with information on navigation obstacles and other moving ships, the problem of avoiding collisions at sea is reduced to a dynamic optimization task with static and dynamic constraints. In this paper a modified version of the EP/N (Evolutionary Planner/Navigator) [13] has been used as a major component of a such decision support system. ϑ EP/N++ computes a safe-optimum path of a ship in a given static and dynamic environment; on the basis of this algorithm, a safe trajectory of the ship in a collision situation is determined. The introduction of a time parameter, together with time-varying constraints representing movable ships, are the main features of the new system. Sample results of ship trajectories obtained for typical navigation situations are presented in the paper.

Key words: evolutionary algorithms, avoiding collisions at sea, Evolutionary Planner/Navigator.

1 Introduction

Finding a safe, anti-collision maneuver is traditionally executed by drawing radar plots based on the observed echoes of the moving objects. The new-built ships are equipped with specialized radar anti-collision systems, Automatic Radar Plotting Aids (ARPA), which facilitate considerably the navigator's work. International Maritime Organization (IMO) has worked out the timetable of the installation of the ARPA systems on the ships built from 1984 [3]. Functions executed by

the ARPA system automate the activities connected with tracking the objects and provide with graphical presentation of the navigational situation. Because of their ability of data processing and display capabilities of the navigational situation on the radar screen, the radar systems ARPA allow the navigator to select reasonable decisions about the maneuver to take. On the basis of the information obtained from the ARPA system (as well as navigator's seamanship and intuition), the final decision on how to act in order to avoid the collision must be made individually by the navigator him/herself.

Apart from adequate preparation for its operation, a proper use of the anti-collision system also requires additional algorithms which would help the navigator in charge to make correct decisions. Recent tendencies in the automation of the ship navigation led to automatic calculations of the anti-collision maneuver, along with simultaneous quantitative assessment of the risk of collision, based on the data obtained from the anti-collision system [1, 4, 5, 6]. Thus the simplest solution seems to be a design of an extension of the conventional anti-collision system ARPA; without interfering in the work of the anti-collision system itself, such a new device would make it possible not only to calculate an anti-collision maneuver, but also to display this maneuver clearly to the navigators steering the ship. The data which describe the moving objects are obtained as the output from the ARPA system, and then are used as input for computing procedures of such a decision supporting system. The system would find a set of effective solutions with respect to the assumed criteria and suggest a compromised decision. The final acceptance of the decision suggested by the system (or the selection of an alternative decision from a set of the effective solutions) is made, of course, by the navigator.

A detailed analysis of models and the synthesis of algorithms for safe, optimum steering was made by Lisowski and Smierzchalski [7]. In their work the problem of determining a safe trajectory as a non-linear programming task was formulated, where a kinematics

model of the own-ship was applied.¹ Another possible approach to this problem is the reduction of the solution space to a finite-dimensional one by creating so-called digitized matrix of permissible maneuvers for a given collision situation and a certain time instant [8]. In [10, 11] the problem of avoiding collisions was formulated as the multi-criterion optimization task. Three separate criteria were used. The attempt to estimate the safe trajectory using classifier systems was presented in [2]. The collision situation was modeled as a fuzzy process with many inputs; for selecting the steering rules the authors made use of a fuzzy classifier system.

The main goal of this paper is to discuss the problem of avoiding collisions at sea from the perspective of an evolutionary process. It seems that the problem of avoiding collisions at sea is, in a sense, similar to the problem of steering a mobile robot. In [13], an evolutionary method (EP/N system) of generating paths of the robot in partially-known environments is presented. Based on the E/PN planning concept, a modified version of the system has been developed which takes into account the specific character of the collision avoiding process. The main innovation of this modified version is the existence of different types of static and dynamic constraints, which reflect the real environment with moving strange ships (targets) and their dynamic characteristics. The evolutionary process which searches for a near-optimum trajectory in a collision situation takes into account a time parameter and the dynamic constraints, which represent strange ships-targets, whose shapes and dimensions depend on assumed safety conditions (i.e., the safe distance between the passing targets, their speed ratio, and bearing).

The paper is organized as follows. The next section defines the problem of planning the own-ship trajectory in collision situations, and discusses the issue of modeling targets in an evolutionary system, whereas section 3 provides examples of planning a safe trajectory. Section 4 concludes the paper.

2 Evolutionary algorithm and the problem of avoiding collisions

In this section we define the problem (environment and constraints), discuss the problem of planning the own-

¹We will use the term “own-ship” to mean the ship for which the path must be generated, and “strange ship” or “target” to mean other vessels in the environment which must be avoided.

ship trajectory in collision situations, and discuss the issue of modeling targets in evolutionary system.

2.1 Definition of environment and constraints

The ship sails in the environment with some natural constraints (e.g., land masses, canals, shallow waters) as well as other constraints resulting from formal regulations (e.g., traffic restricted zones, fairways, etc). It is assumed that these constraints are stationary and that they can be defined by polygons — like the way the electronic maps are created. When sailing in a stationary environment, the own ship meets other sailing strange ships/targets (some of these targets constitute a collision threat).

The degree of the collision threat with dangerous targets is not constant and depends on the approach parameters: D_{CPA} (Distance at Closest Point of Approach) and T_{CPA} (Time of Closest Point of Approach), as well as on the speed ratio of both ships, and the distance and bearing of the target.

It is assumed that the dangerous target is the target that has appeared in the area of observation² and can cross the estimated course of the own ship at a dangerous distance. In the evolutionary task, the targets threatening with a collision are interpreted as moving dangerous areas having shapes and speeds corresponding to the targets determined by the ARPA system. The shapes of these dangerous areas depend on the safety conditions: an assumed safe distance, D_{safe} ³, speed ratio, and bearing.

Figure 1 displays the model of the environment, where:

- fixed navigation constraints are modeled using convex and concave polygons,
- targets are modeled as moving hexagons,
- the dimensions of the own ship are neglected due to small length of the own ship with respect to the maximum length of the areas representing the targets.

2.2 Planning the trajectory in collision situation

According to transport plans, the own ship should cover a given route R_0 in some assumed time. On

²Ranges of 5–8 N. miles in front of the bow, and 2–4 N. miles behind the stern of the ship are assumed. Their actual values depend on the assumed time horizon.

³A safe distance is selected by the operator depending on the weather conditions, the sailing area, and the speed of the ship.

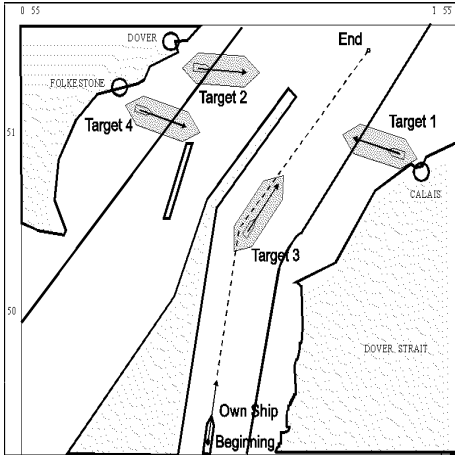


Figure 1: Navigation situation in Dover Straits

the other hand, it has to move safely down a given trajectory, i.e., it must avoid navigation obstacles and cannot come too close to other targets. Estimation of a trajectory of the ship in a collision situation represents a difficult trade-off between a necessary deviation from a given course and the safety of sailing. Hence it is a multi-criterion planning problem, which takes into account the safety and economy of the ship motion.

The estimation of the own ship trajectory in the collision situation consists of determining a path, S , as the part of the given route R_0 , from the present location (starting point) $(x_0, y_0) \in R_0$ to the actual end point $(x_e, y_e) \in R_0$. This path has the form of a sequence of elementary line segments s_i ($i = 1, \dots, n$), linked with each other in turning points (x_i, y_i) . The choice of the actual starting and end point, depends on an assumed sensible horizon and is made by the operator. The boundaries of the environment are defined as

$$X = \{\vec{x} \in R^2 : a_i \leq x_i \leq b_i \text{ for } i = 1, 2\}; \quad (1)$$

O_stat_i ($i = 1, \dots, k$) and $O_dyn_i(t)$ ($i = k + 1, \dots, l$) represent the sets of static and dynamic constraints, respectively. The space $S_safe(t)$ of safe (anti-collision) paths is defined as

$$S_safe(t) = X - \bigcup_{i=1}^k O_stat_i - \bigcup_{i=k+1}^l O_dyn_i(t). \quad (2)$$

In other words, a path S is safe (i.e., it belongs to the set of safe paths $S_safe(t)$) if any segment s_j ($j = 1, \dots, n$) of S stays within the limits of environment X , does not cross static constraints O_stat_i , and at the time instances t determined by the current locations of the own ship, does not come in contact with moving areas $O_dyn_i(t)$ representing targets. Paths which cross the restricted areas generated by static

and dynamic constraints are called unsafe, or dangerous paths.

The task of estimating the own ship trajectory in a collision situation (so-called the *steering goal*) is performed as an evolutionary search for safe paths in the permissible space X , with subsequent selection of a near-optimum path S_opt from the set S_safe with respect to the fitness function (defined by the path cost).

2.3 Evolutionary system: ϑ EP/N++

In [13] a description of an evolutionary algorithm, Evolutionary Planner/Navigator (EP/N) was provided as a novel approach to path planning and navigation. The system unified off-line planning and on-line planning/navigation processes in the same evolutionary algorithm which (1) accommodated different optimization criteria and changes in these criteria, (2) incorporated various types of problem-specific domain knowledge, and (3) enabled good trade-offs among near-optimality of paths, high planning efficiency, and effective handling of unknown obstacles. However, all reported experiments [13] were limited to static obstacles only. Consequently, the speed of the controlled robot was of no significance.

Based on the E/PN planning concepts, a modified version of the system (ϑ EP/N++) has been developed which takes into account specific character of the collision avoiding process. The new system preserves the original structure of the EP/N (e.g., it is also a steady-state system where two populations separated by one generation differ at most by a single individual, it has the same set of eighth variation operators, etc). However, there are differences. The main innovation of the modified version is that it processes both static and dynamic constraints, which reflect the real environment of fixed navigation constraints as well as moving strange ships (targets), whose shapes and dimensions depend on assumed safety conditions (the safe distance between the passing targets, their speed ratio, and bearing). There are several consequences of this modification and the new system, ϑ EP/N++, differs from the original EP/N in several aspects:

- it processes dynamic constraints as well as static ones,
- it introduces the concept of time, which is essential while dealing with movable obstacles, and
- it allows the own ship to change its speed.⁴

⁴Experiments where the own-ship changes its speed are described in the full version of this paper [12].

Consequently, some changes were made with respect to path representation (e.g., each path segment includes a value for speed for this segment) and operators. The evaluation function was changed as well: the concept of the best path, apart from distance, smoothness and clearance, includes also the time to traverse the path. It was also necessary to develop new procedures for modeling dynamic obstacles for a given path and calculating a position of the own ship with respect to these obstacles.

Some operators (e.g., “repair”) defined for the original EP/N [13] modify the path by selecting an infeasible segment and then pulling the segment around the intersecting obstacles, thus repairing the selected segment. Note, that the action of the counterpart operator in ϑ EP/N++ is the same, however, it repairs infeasible segments in the presence of dynamic constraints (it pulls the segment around static and dynamic obstacles).

2.4 Modeling of targets in evolutionary environment

The own ship is assumed to move with a uniform speed (along a safe path S) from the starting point (x_0, y_0) to the end point (x_e, y_e) , and at the initial instant t_0 the motion of the strange ships (targets) is defined as uniform. For each target, its motion is represented by the following parameters: bearing, distance, speed, and course, estimated by the ARPA system. Each path (individual) is first generated in a random way. Next, each path is evaluated. To determine whether a path is safe, the path is examined with respect to the set of static and dynamic constraints. The instantaneous locations of the dynamic areas with respect to the evaluated path depend on time t_0 , determined by the first crossing point (x_{int}, y_{int}) between the own ship’s path S and the trajectory of the target (in the figure 2, this crossing point is the point of the biggest collision threat). Thus, in this figure, three positions of a target are displayed for three paths. In further figures of this paper, we display the positions of the target with respect to the best path in the population).

Having known the length of the line segment from the starting point (x_0, y_0) to the crossing point (x_{int}, y_{int}) and assuming that the own ship will keep moving with the uniform speed ϑ , it is possible to determine time t_0 which the own ship needs to cover this distance.

After time t_0 , the instantaneous location of the target with respect to the own ship is modeled as a dangerous area of hexagonal shape. The detailed shape and dimensions of the hexagon depend on the safety conditions given by the operator.

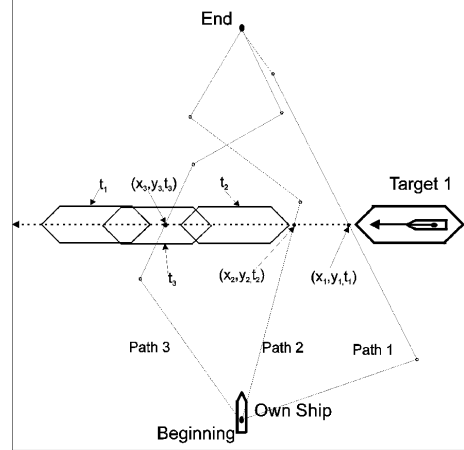


Figure 2: Crossing paths and dangerous areas attributed to them

After the paths are evaluated, selected paths are modified by specialized set of operators (for details, see [13]).

The values assumed in the paper are the following:

- the distance in front of the bow which guarantees avoidance of the collision is equal to $3 \cdot D_{safe}$ (in practice, safe distance D_{safe} is taken from the range between 0.5 and 3.0 nautical miles),
- the distance behind the stern is equal to D_{safe} ,
- the width of the dangerous area on each side of the own ship is chosen with the preference of the ship’s passage behind the stern of the target, which depends on the course and bearing of the target.

3 Planning a safe trajectory

The operation of the system has been examined for a number of collision situations. The first example (Figure 3) shows the situation when the own ship approaches a target on its right side. The population consisted of 10 individuals (paths), and the system converged after 300 generations (3 seconds⁵ of CPU time). It is clear that the own ship, steering along the developed trajectory, will pass the target safely, passing it behind the stern.

The second example (Figure 4) represents two targets sailing with opposite courses on the right and left sides of the own ship. In the evolutionary process, two dynamic constraints are taken into account. The population consisted of 30 paths, and the system converged

⁵All calculations were done on a Sun Ultra-Enterprise 4000 with 2 processors UltraSPARC 167MHz.

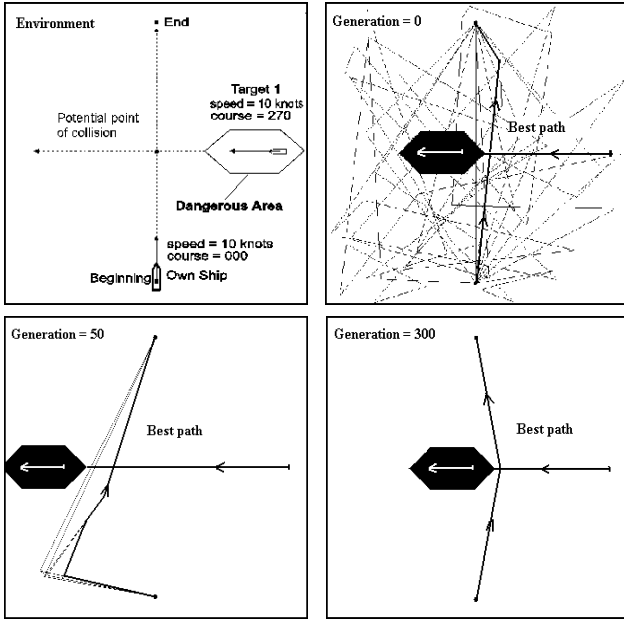


Figure 3: Paths evolution for the case of approaching one target

after 450 generations (5 seconds of CPU time). The estimated trajectory secures the passage of the own ship behind the sterns of the targets 1 and 2 on the left and on the right side, respectively.

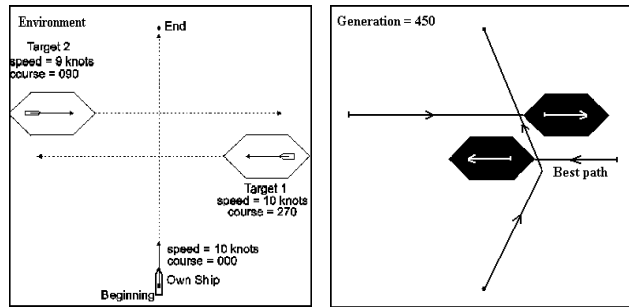


Figure 4: Paths evolution for two targets being approached

The third example (Figure 5) models an environment with static navigational constraints, and with three targets. This example is quite characteristic for sailing in narrow passages. The population consisted of 10 paths, and the system converged after 1100 (28 seconds) generations. The estimated trajectory secures a safe passage.

The fourth example (Figure 6) represents two targets and an environment with additional static navigational constraints. The population consisted of 30

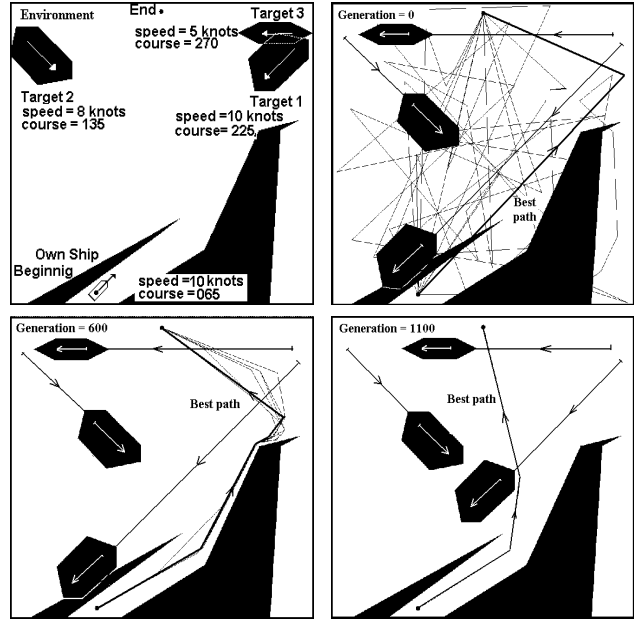


Figure 5: Path evolution for the case of approaching 3 targets in the presence of static navigation constraints

paths and the system converged 800 generations (23 seconds). The estimated trajectory secures the passage of the own ship behind the sterns of targets 1 and 2.

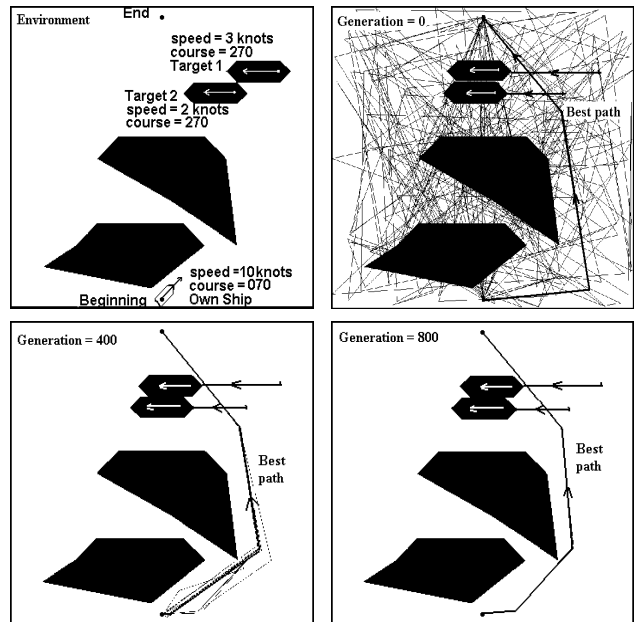


Figure 6: Path evolution for the case of approaching two targets in the presence of static navigation constraints

4 Conclusions

The evolutionary method of estimating the safe and optimum passing path, being the own ship's trajectory in the environment with static and dynamic constraints, represents a new approach to the problem of avoiding collisions at sea. A number of preliminary tests allows us to formulate the following conclusions:

- evolutionary algorithms can be effectively used for solving the problem of avoiding collisions at sea, where an environment is modeled as a set of polygons representing static and dynamic constraints,
- the task of evolutionary estimation of the own ship trajectory in a collision situation is reduced to an adaptive search in a set of safe paths S_{safe} , with subsequent selection of the optimum trajectory with respect to the fitness function,
- strange ships (targets) are modeled in the evolutionary process as dynamic constraints, a moving restricted area having a hexagonal shape. The detailed shape and dimensions of the hexagon depend on safety conditions and parameters of motion entered by the operator.

The introduction of additional elements to the program in order to include other environment changes does not impose any significant problems in the evolutionary path planning, and, undoubtedly, make the process more similar to real navigation situations. Each newly occurred situation can be in a natural way added to the operational diagram of the evolutionary algorithm. Further research plans in this area are directed at estimating the ship trajectory when the strange ships (targets) change the parameters of their motion in an unforeseen manner while the own ship moves along its trajectory.

Acknowledgements

The research reported in this paper was partially supported by the grants 8 T11C 010 10 and 8 T11C 027 10 from the Polish State Committee for Scientific Research.

References

- [1] Davis P.V., Dove M.J., and Stockel C.T., Computer Simulation of Multiship Encounters. *Journal of Navigation*, Vol.35, No.2, 1982.
- [2] Furuhashi T., Nakaoka K., and Uchikawa Y., A Study on Classifier System for Finding Control Knowledge of Multi-Input Systems, (F. Herrera and J.L. Verdegay, Editors), *Genetic Algorithms and Soft Computing*, Physica-Verlang, 1996.
- [3] IMO Preference Standards for Automatic Radar Plotting Aids (ARPA). Resolution A. 422 (XI), Nov. 1979.
- [4] Jones K.D., Decision Making when Using Collision Avoidance System. *Journal of Navigation*, Vol.31, No.2, 1978.
- [5] Lamb W.G.P., The Calculation of Marine Collision Risks. *Journal of Navigation*, Vol.38, No 4, 1985.
- [6] Lisowski J., A Simulation Study of Various Approximate Models of Ships Dynamics in the Collision Avoidance Problem. *Foundation of Control Engineering*, Vol.10, No.2, 1985.
- [7] Lisowski J. and Smierzchalski R., Assigning of Safe and Optimal Trajectory Avoiding Collision at Sea. 3rd IFAC Workshop Control Applications in Marine System, Trondheim-Norway 1995.
- [8] Lisowski J. and Smierzchalski R., Methods to Assign the Safe Maneuver and Trajectory Avoiding Collision at Sea. 1st International Conference Marine Technology. Szczecin 1995.
- [9] Michalewicz Z., *Genetic Algorithms + Data Structures = Evolution Programs*. Spriger-Verlang, 3rd edition, 1996.
- [10] Smierzchalski R., The Application of the Dynamic Interactive Decision Analysis System to the Problem of Avoiding Collisions at the Sea, (in Polish) 1st Conference Awioniki, Jawor, Poland 1995
- [11] Smierzchalski R., The Decision Support System to Design the Safe Maneuver Avoiding Collision at Sea. ISAS'96, Orlando, USA, 1996.
- [12] Smierzchalski R. and Michalewicz, Z., Adaptive Modeling of a Ship Trajectory in Collision Situations at Sea. Submitted for publication, 1998.
- [13] Xiao J., Michalewicz Z., Zhang, L., and Trojanowski, K., Adaptive Evolutionary Planner/Navigator for Mobile Robots, IEEE Transactions on Evolutionary Computation, Vol.1, No.1, 1997.