RAFAŁ SZŁAPCZYŃSKI
Gdansk University of Technology

DETERMINING THE OPTIMAL COURSE ALTERATION MANOEUVRE IN A MULTI-TARGET ENCOUNTER SITUATION FOR A GIVEN SHIP DOMAIN MODEL

ABSTRACT

The paper introduces a new numerical deterministic method of finding the necessary course alteration manoeuvre in a multi-target encounter situation for a given ship domain model. Its simplicity, flexibility and low computational complexity make this algorithm a useful component for real-time systems, where processing time is of importance. The algorithm is presented explicitly, so that it could be directly applied in any on-board collision avoidance system or VTS system. Example results of its use are also provided.

Keywords:
course manoeuvre, multi-target situation, collision avoidance.

INTRODUCTION

The ways of dealing with multi-target encounter situations can be roughly divided into two groups: algorithms determining collision-avoidance manoeuvres and methods of determining safe trajectories. Some of the methods of determining safe trajectories include the algorithms determining collision-avoidance manoeuvres as their components. These methods (presented in [1, 4] among others) usually assume that a multi-encounter situation can be reduced to a sequence of single-encounters. Unfortunately this is not always true. If the times remaining to meeting with different targets are similar, the encounter situations have to be handled simultaneously instead of sequentially. This task may be solved by the algorithm presented in the paper. This algorithm has the additional quality of supporting any convex ship domain. The latter feature is a result of applying a special collision risk measure introduced by the author [3]. This measure is called the approach factor f_{min}. It is defined as the scale factor of the largest domain-shaped area that is predicted to remain free from other ships throughout the whole encounter situation.

DETERMINING THE OPTIMAL COURSE ALTERATION MANOEUVRE – THE ALGORITHM DESCRIPTION

In brief the algorithm works as follows:

1. For every target, to whom the own ship is obliged to give way, the most dangerous own courses (resulting in the minimal value of the approach factor f_{min}) are determined. These courses will be referred to as the closest approach courses.
2. For each of the closest approach courses it is checked whether the f_{min} value for this course is lesser than one. If it is, then the range of forbidden own courses is determined around this course.
3. A safe own course that is closest to the desired course and compliant with COLREGS is determined – it is the optimal course.
4. A time for which the chosen course has to be kept, before it can be changed to the course heading directly to the destination point, is determined.

The above listed four subsequent steps of the algorithm are thoroughly presented in subsections 1 – 4 respectively.

1. Determining the closest approach courses for a given target

In most cases the closest approach course is equal to a crash course, that is to a course resulting in both *DCPA* and f_{min} being equal to zero. However, if the own speed is lesser than the speed of a given target, there may not be a crash course, but the target's (or the own ship's) domain may still be violated. Therefore the term of the closest approach course is introduced and it might be thought of as a certain generalization of a crash course term. The closest approach course is a course resulting in the minimal value of the approach factor f_{min}.

Narrowing the search range

To improve the algorithm performance it is useful to narrow the course range, where the closest approach courses are searched for. For practical reasons the closest approach course is only of interest if the f_{min} value associated with it is lesser than one (domain violation). The f_{min} value for a given domain model can only be lesser than one if the condition $DCPA < a$ holds, where a is the distance between a ship and the most distant point on its domain boundary. Thus it is enough to find the minimal f_{min} value and the corresponding course for each of those course ranges, where $DCPA < a$ holds. These course ranges can be determined by finding the boundary courses for which $DCPA = a$. The formula for determining these courses as kinematical manoeuvres is as follows [2]:

$$tg\frac{\psi}{2} = \frac{AV \pm \sqrt{V_1^2(A^2+1) - B^2)}}{B - V_1}, \tag{1}$$

where:

$$A = \frac{xy \pm a\sqrt{R^2 - a^2}}{x^2 - a^2}; \tag{2}$$

$$B = AV_2 cos\psi_2 - V_2 sin\psi_2; \tag{3}$$

V_1 – the speed of the own ship;
V_2 – the speed of the target ship;
ψ_2 – the course of the target ship;
x_r, y_r – components of the target ship's position in relation to the own ship.

Equation (1) only has solutions if the following two conditions hold:

$$V_1^2 \geq \frac{B^2}{A^2+1}; \tag{4}$$

$$V_1(x sin\psi + y cos\psi) > xV_2 sin\psi_2 + yV_2 cos\psi_2, \tag{5}$$

where (5) results from the condition $TCPA > 0$ (only future domain violations are of interest).

Depending on the obtained values of A and B there may be up to four boundary courses for which $DCPA = a$, that is up to two course ranges for which $DCPA < a$.

Searching for the closest approach course within a given range

Once a range of courses has been determined, the closest approach course is searched for within this range by means of the bisection algorithm. In this phase the own ship's dynamics is taken into account so as to determine the closest approach course precisely. The algorithm of searching for the closest approach course is presented in fig. 1.

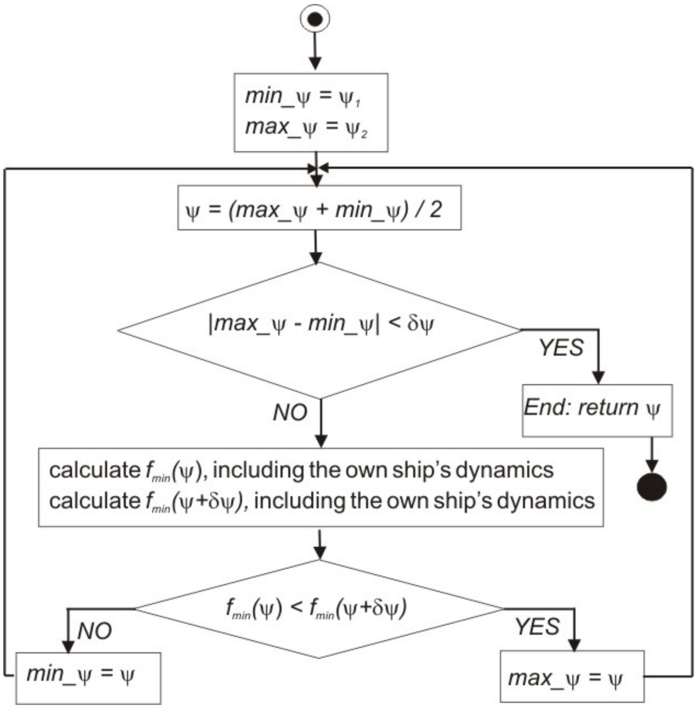

Fig. 1. Searching for the closest approach course within a given range of courses

The algorithm takes the following input parameters:

- the previously determined range of courses (ψ_1, ψ_2);
- the accuracy of finding the closest approach course $\delta\psi$;
- the own ship's delay time given as a function of course alteration manoeuvre $t_d(\Delta\psi)$.

2. Determining the range of forbidden courses for a given closest approach course

Once a closest approach course has been found, the boundaries of the forbidden sector of the own courses are determined. In fig. 2. and fig. 3. the algorithms of determining the values of the starboard and port board boundaries of the forbidden courses sector respectively are presented. Apart from the standard parameters of both ships (positions, courses, speeds) these algorithms take two additional parameters:

- the previously found closest approach course ψ;
- the own ship's delay time as a function of course alteration manoeuvre $t_d(\Delta\psi)$.

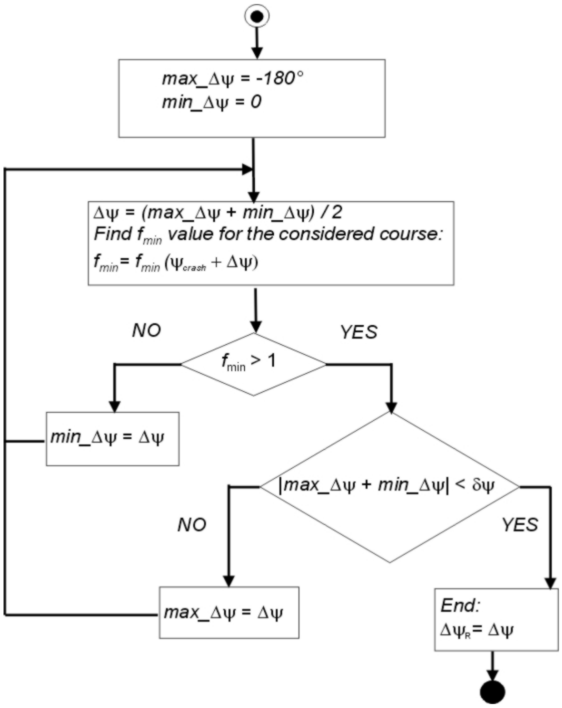

Fig. 2. The algorithm determining the starboard boundary of the forbidden sector of the own courses

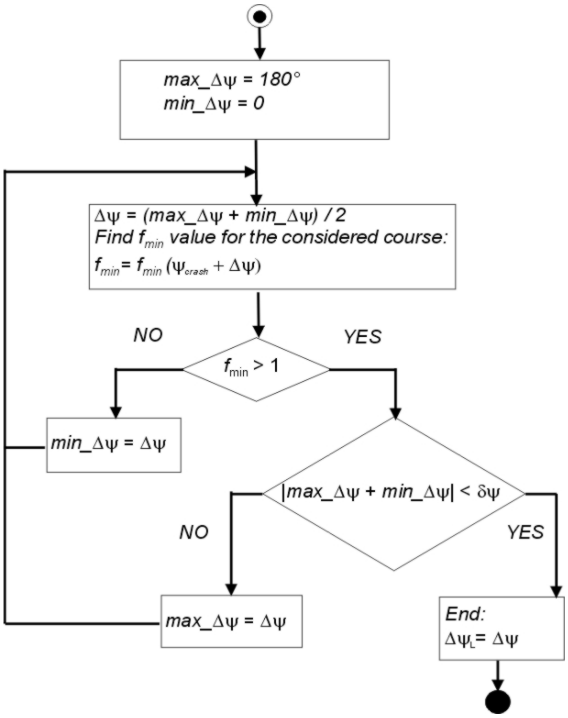

Fig. 3. The algorithm determining the port board boundary of the forbidden sector of the own courses

Different results of forbidden course sectors obtained for different ship domains are shown in figures below. In fig. 4 and fig. 5. the examples of forbidden courses for a circle-shaped and ellipse-shaped domain respectively are shown.

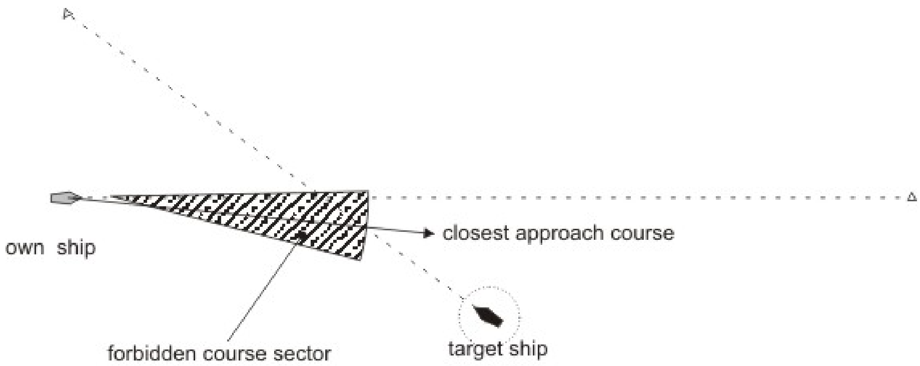

Fig. 4. The sector of forbidden own courses for an encounter with a single target for a circle-shaped domain

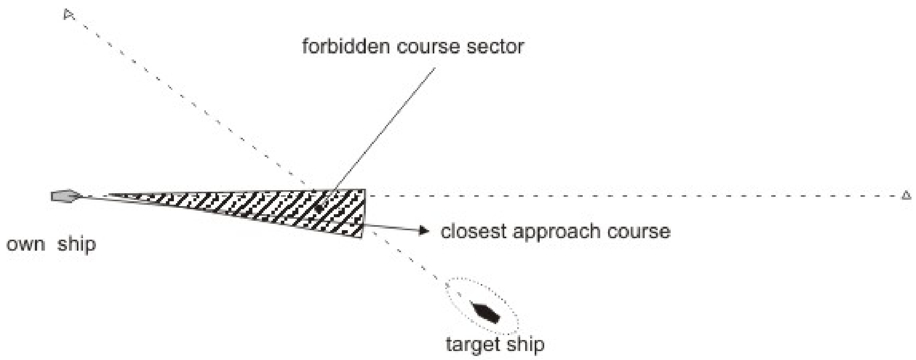

Fig. 5. The sector of forbidden own courses for an encounter with a single target for an ellipse-shaped domain

3. Determining the optimal course alteration manoeuvre for a multi-target encounter

The optimal course alteration manoeuvre for a multi-target encounter is the alteration which fulfils the following conditions:

– it is compliant with COLREGS, that is, it is equal to zero or no lesser than 15 degrees;
– it guarantees a safe passage – ship domains will not be violated;

– the altered course is as close as possible to the desired course, that is, to the course directed towards the destination point (or the next turning point of the predetermined route).

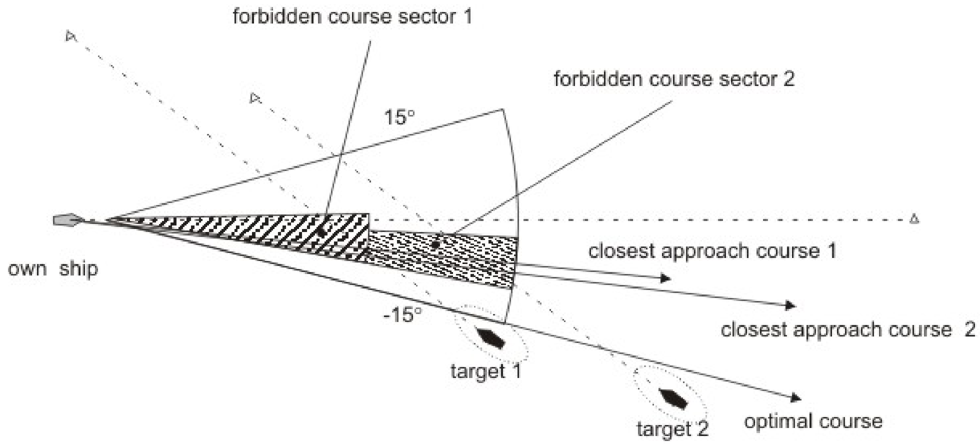

Fig. 6. Overlapping sectors of forbidden own courses for an encounter
with two targets and the resulting optimal course of the own ship

The first two conditions are treated as the constraints, the third one – as the optimisation criterion. The safety constraints are given as a sequence of forbidden course sectors (possibly overlapping) obtained for all closest approach courses (fig. 6). The COLREGS condition might also be represented by means of forbidden course sectors. It is enough to add two extra course ranges to the forbidden course sectors list: (current course –15 degrees, current course) and (current course, current course +15 degrees). This will guarantee that only course alterations larger or equal to 15 degrees will be taken into account. As for the optimisation criterion, the optimal altered course has the following quality: it is either equal to the desired course or equal to one of the boundaries of the forbidden course ranges. Therefore it might be simply determined by the following procedure:

1. Check whether the optimal course is equal to the desired course. This can be done by checking whether the desired course lies outside of all forbidden course sectors. If it does – it is the solution to the task. If it does not - the next step has to be applied.
2. Make a list of candidate courses in the following way. For all boundary courses of forbidden course sectors check whether a current boundary course lies outside of all other forbidden course sectors. If it does – insert it into the candidate list.

3. From the candidate list choose the one which is closest to the desired course. If there are two courses equally close to the desired course: one meaning an alteration to the starboard, the other to the port board – choose the one which crosses astern of the targets (this usually means choosing the alteration to the starboard).

4. Determining for how long the altered course has to be kept

Determining for how long the altered course has to be kept, before all considered targets are safely passed, is done as follows:

1. Arrange the targets in the ascending order of their t_{min} times (times remaining to reaching f_{min} values).
2. For each of the prioritised targets determine the appropriate time by means of the bisection algorithm. The time value determined for the current target is passed to the algorithm determining the necessary time for the next target as the lower time boundary. This guarantees that the time value, which is to be determined for the next target, will not be lesser than the time value that has already been determined for the previous target.
3. Choose the value that has been determined for the last target.

SIMULATION EXAMPLES

An example trajectory, which is a result of a course alteration manoeuvre avoiding collision with two targets is exemplified by fig. 7. The simulation data for this example are provided in table 1, table 2 and table 3. The numbers in brackets in fig. 7. denote different phases of the situation.

Table 1. Positions, courses and speeds of the own ship and the target ships

	Speed [knots]	**Course** [degrees]	**Position coordinates at the start time**		**Coordinates of the destination point**	
			x [n.m.]	y [n.m.]	x [n.m.]	y [n.m.]
Own ship	15	90	3	6	17	6
Target 1	8	305.5	10	4		
Target 2	8	305.5	13	3		

Table 2. The assumed parameters of the Fuji domain

Major semi-axis [n.m.]	Minor semi-axis [n.m.]
0.7	0.3

Table 3. The additional configuration parameters

Minimal acceptable course alteration	Maximal acceptable course alteration	Decision time	Turning speed
15°	60°	3 minutes	1° / second

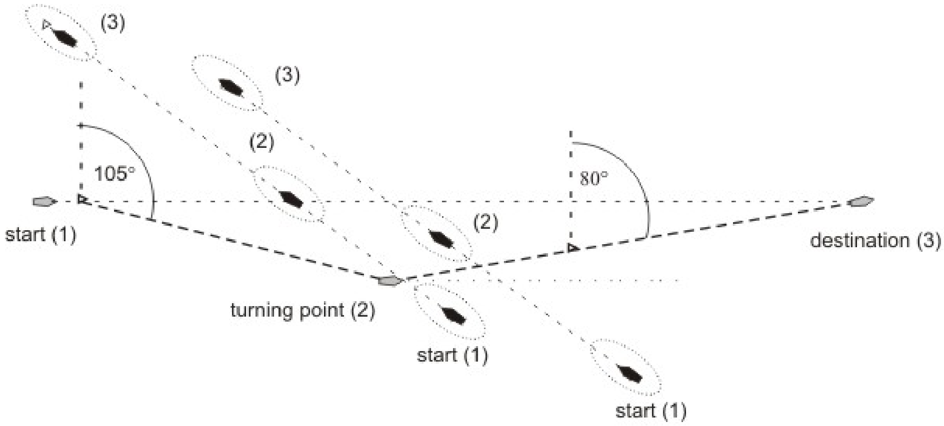

Fig. 7. A trajectory which is a result of a single course alteration manoeuvre avoiding collision with two targets

The trajectory in fig. 7 is not only a safe one but for this particular situation it would be also close to the optimal one, because its way loss is insignificant (only 2% of the original trajectory length). However, if the t_{min} values (the times remaining to reaching the f_{min}) varied considerably for different targets, the own ship would have to keep the altered course for a much longer time thus increasing the way loss. Therefore in the most general case the presented algorithm should rather be used as a component inside a more complex method for finding safe trajectories – a method which would include planning multiple collision-avoidance manoeuvres when necessary. An example of applying the algorithm within a method of determining safe trajectories is shown in fig. 8. The ship parameters and domain dimensions (Coldwell domain is used this time) are given in table 4 and table 5 respectively.

Table 4. Positions, courses and speeds of the own ship and the target ships

	Speed [knots]	**Course** [degrees]	**Position coordinates at the start time**		**Coordinates of the destination point**	
			x [n.m.]	*y* [n.m.]	*x* [n.m.]	*y* [n.m.]
Own ship	15	90	3	6	17	6
Target 1	8	305.5	10	4		
Target 2	8	305.5	13	3		
Target 3	8	310.5	19	1		
Target 4	8	310.5	20	0		

Table 5. The assumed parameters of the Coldwell domain

Major semi-axis [n.m.]	Minor semi-axis [n.m.]	Domain centre moved from the ship's position towards starboard [n.m.]	Domain centre moved from the ship's position towards bow [n.m.]
0.8	0.4	0.1	0.2

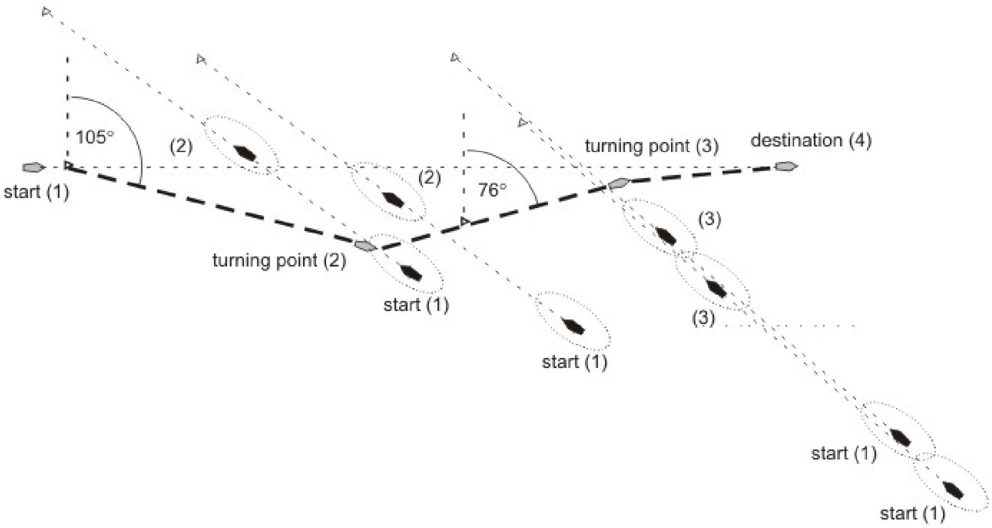

Fig. 8. A safe trajectory determined by a method that incorporates the presented algorithm

CONCLUSIONS

The algorithm presented in the paper determines the optimal dynamical course alteration manoeuvre avoiding collisions with several targets. It is compliant with COLREGS and it supports any convex ship domain due to applying the author-designed collision risk measure. It is also very fast because of bisection algorithms that have been used here. The computational complexity of the main algorithm is a logarithmic function of the considered time range and linear function of the number of the targets. In some situations the algorithm may be directly applied as a stand-alone tool. In others – it can be utilized inside a method of determining safe trajectories. As has been shown in the paper – both cases result in trajectories which are safe as well as economical.

REFERENCES

[1] Davis P. V., Dove M. J., Stockel C. T., A Computer Simulation of Multi-Ship Encounters, Journal of Navigation, vol. 35, Cambridge 1982.

[2] Lenart A. S., Collision threat parameters for a new radar display and plot technique, The Journal of Navigation, vol. 36, Cambridge 1982.

[3] Szłapczyński R., A Unified Measure of Collision Risk Derived from the Concept of a Ship Domain, Journal of Navigation, vol. 59 iss. 03, Cambridge 2006.

[4] Śmierzchalski R., The synthesis of methods and algorithms supporting navigational decisions in collission situations at sea, Works of Gdynia Maritime University, Gdynia 1998 (in Polish).

Received August 2006
Reviewed January 2007