TransNav
http://www.transnav.eu
the International Journal on Marine Navigation and Safety of Sea Transportation
Volume 9
Number 3
September 2015

DOI: 10.12716/1001.09.03.05

Trajectory Planning with Negotiation for Maritime Collision Avoidance

S. Hornauer & A. Hahn
University of Oldenburg, Germany

M. Blaich & J. Reuter
University of Applied Sciences Konstanz, Germany

ABSTRACT: The problem of vessel collisions or near-collision situations on sea, often caused by human error due to incomplete or overwhelming information, is becoming more and more important with rising maritime traffic. Approaches to supply navigators and Vessel Traffic Services with expert knowledge and suggest trajectories for all vessels to avoid collisions, are often aimed at situations where a single planner guides all vessels with perfect information. In contrast, we suggest a two-part procedure which plans trajectories using a specialised A* and negotiates trajectories until a solution is found, which is acceptable for all vessels. The solution obeys collision avoidance rules, includes a dynamic model of all vessels and negotiates trajectories to optimise globally without a global planner and extensive information disclosure. The procedure combines all components necessary to solve a multi-vessel encounter and is tested currently in simulation and on several test beds. The first results show a fast converging optimisation process which after a few negotiation rounds already produce feasible, collision free trajectories.

1 INTRODUCTION

In the last decades marine traffic increased significantly and consequently did the collision risk for vessels. A lot of assistance systems like GPS, ARPA, AIS or ECDIS were introduced to prevent ship accidents, however, their number is still on a constant, high level. Only groundings have decreased slightly. About half of the accidents are caused by human factors and take place in congested areas, during port approach or in harbours, as stated by reports of the Baltic Marine Environment Protection Commission - Helsinki Commission (HELCOM).

On-board and onshore assistance systems, as well as future collision avoidance systems, can benefit from knowledge about suggested trajectories for all vessels in an encounter, which are not only locally optimised but also coordinated between vessels to provide guidance in critical traffic situations. Trajectory planning methods are used already to plan manoeuvres or guide single vessels in two-vessel encounters. An overview of the most important approaches is given by Statheros et al. (2007) and Tam et al. (2009). Some collision avoidance approaches are limited to two-vessel encounters or estimate evasive manoeuvers for the own vessel only. For this work the focus is on approaches for n-vessel scenarios optimising all the trajectories of the involved vessels. Heuristic optimization algorithms are mainly used to solve this problem. One of the first and most promising approaches using an evolutionary algorithm is presented by Smierzchalski (1999). Many other approaches use heuristic methods like fuzzy-logic, neural networks or ant algorithms. However, the most sophisticated approach is presented by Szlapczynski (2011) and

Szlapczynski, R. & Szlapczynska J. (2012a, b). They use an evolutionary algorithm with specialised operators to shape the convergence of the optimisation. In Szlapczynski (2012, 2013) this approach is extended to the use within traffic separation schemes (TSS). In order to limit the variety of individuals of a population during evolutionary optimisation Szlapczynskis approach generates tracks, already partially valid within a TSS after which a number of defined violation are penalised using a specialised fitness function. A further improvement of this algorithm is made for tracks in restricted visibility according to the rule 19 of the Collision Avoidance Regulations (COLREGs) and is presented in Szlapczynski (2015).

All these approaches are more suitable for onshore applications like Vessel Traffic Services (VTS) because the information about the current traffic situation has to be complete. However, for an on-board usage the limited common situation awareness limits global planning approaches with n-vessels. Suggested trajectories for several vessels must not only take information into account, which are often hard or impossible to acquire for a single, planning observer, but also satisfy a number of constraints imposed by COLREGs and the vessel's dynamic model.

1.1 *Methodology*

This paper proposes a staged procedure to find and distribute and near optimal set of trajectories for a number of vessels. During the procedure, independent components for motion prediction, trajectory generation and trajectory negotiation are used to optimise all trajectories according to a global performance measure (safety / efficiency). First, the motion prediction estimates the future motion of the other vessels based on their trajectories or on motion models if a trajectory is unavailable after which the trajectory generation algorithm plans a new avoidance trajectory in a local area, given the locally available information which will increase during the negotiation. Finally the trajectory negotiation component is used to find a global optimal solution as fast as possible while the amount of data that needs to be transferred is kept as little as possible. A beneficial side effect is that the intentions of all vessels in a critical situation are known early in the process.

Even though new communication infrastructures are being developed, which will provide advanced communication architectures in the future, i.e. by Mu et al. (2011), the bandwidth is still considered as limited in the near future. Therefore, the explicit communication of all missing information is regarded too costly on sea leading to our approach of exchanging trajectories only. These trajectories contain implicit information about preferences, ship's capabilities, parts of the ship's value function and the environment without needing to explicitly disclose all those information. Furthermore, decentralised algorithms are considered as computationally favourable and can find Pareto-Optimal solutions without needing to know the value function of trajectories for other negotiators, as shown by Heiskanen (1999).

The described procedure is modelled and tested in a simulation to compare a number of different trajectory generation and negotiation approaches and afterwards evaluated in different scenarios using simulation and in several maritime test-beds at Lake Constance and the German Bight.

2 TRAJECTORY DEFINITION

A trajectory is the exact path of the vessel over ground; similar to tracks for vojage planning but including exact turns. Therefore a Trajectory is defined by j waypoints $\mathbf{w}_0 .. \mathbf{w}_{j-1}$ and a speed v to travel alonge the trajectory. To achive continuous turning rates the segments between the waypoints can be interpolated using fifth order Bézier curves. Thus, the trajctory consist of $j-1$ Bézier curves. Figure 1 shows an example of a trajectory defined by four waypoints $\mathbf{w}_0 .. \mathbf{w}_3$.

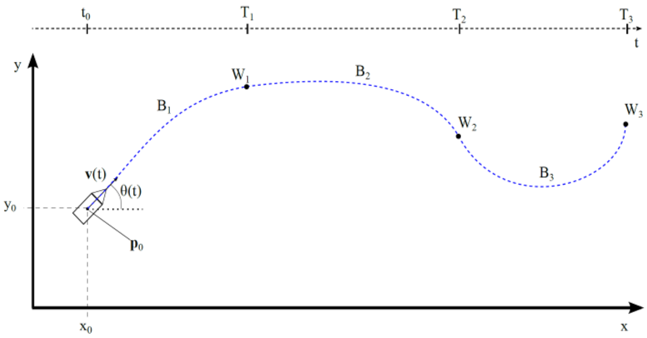

Figure 1. A vessel trajectory defined by four waypoints. The first waypoint is the current vessel position. The segments between the waypoints are interpolated by three Bézier curves.

The Bézier curves used for the interpolation are parametric curves $\mathbf{B}_k(c)$. A Bézier curve of order n used for the k-th path segement is defined by $n+1$ control points $\mathbf{P}_0 .. \mathbf{P}_n$ and a Bernstein polynomial:

$$B_k(c) = \sum_{i=0}^{n} b_{i,n}(c) P_{i,k}, \qquad c \in [0,1], \quad 0 < k < j \tag{1}$$

with $b_{i,n}$ the i-th Bernstein polynomial of order n. The interpolation is applied between the waypoints $\mathbf{w}_{k-1}$ and $\mathbf{w}_k$. The length of the curve for the k-th segment is calculated by:

$$L_k(c) = \int \sqrt{x'(c)^2 + y'(c)^2}\, dc \tag{2}$$

To calculate the position of the vessel alonge the trajectory at any time the parameter c is defined as a function of the normalised time

$$c(t) = \frac{t - T_{k-1}}{\Delta t_k} \tag{3}$$

with T_k the time when waypoint $\mathbf{w}_k$ is reached and $\Delta t_k = T_k - T_{k-1}$ the time required to pass the segment, which is calculated appling the track speed v to the length of the curve $L_k(c)$. Using equation 1 and 3 the position of the vessel on the trajectroy at any time $\mathbf{p}(t)$ can be calculated by:

$$p(t) = \sum_{i=0}^{n} b_{i,n}\left(c(t)\right) P_{i,j}, \quad 0 < k < j \tag{4}$$

This interpolation method allows the definition of a complete vessel trajectoy $R(W,v)$ using a set of waypoints $\mathbf{W} = \{\mathbf{w}_0 .. \mathbf{w}_{j-1}\}$ and the track speed v.

3 MOTION PREDICTION

To identify possible collisions and for the avoidance algorithms, knowledge about future motions of the own and other vessels is necessary. Possible information about the vessel's current state can be received e.g. by an AIS/ARPA system. For motion prediction, necessary information are position, speed, direction of motion and the length of the vessel. Further optional information like the rate of turn or a route plan consisting of waypoints can be used to improve the performance of the prediction. Thus, an approach considering all available information is proposed in this work. If waypoints and a track speed are available, this information is used to generate a trajectory, and the *trajectory following* (TF) method described in section 3.3 is used to estimate the vessel's position, traveling along this trajectory. If waypoints for a vessel are unavailable, the future motion of the vessel is predicted using two simple motion models. One model is used for straight line motion and another model is used for circular motion. Due to the imprecise estimation of vessels acceleration and the unknown desired speed during an acceleration or deceleration manoeuvre the assumption of a constant velocity is used for both models.

For straight line motion the *constant velocity* (CV) model is used. This non-holonomic model allows movements with very low velocity uncertainties. This leads to a good state estimation for straight line motion but for manoeuvring motions the state estimation impairs. The model for a circular motion uses an approximately constant turn rate to define the orientation change of a vessel and is called *constant turn rate and velocity* (CTRV) model. The velocity uncertainties in this model are larger than for the constant velocity model. This improves the state estimation for manoeuvring motions but impairs the accuracy of a straight line motion estimation. The models are based on experience from the target tracking community and explained in detail by Li & Jilkov (2003). For two-vessel encounter scenarios, Schuster et al. (2014) presented an approach to generate an evasive trajectory for the own vessel, using those two models for the motion prediction of the other vessel.

3.1 *Constant Velocity Model (CV)*

For the constant velocity model the longitudinal and lateral position $\mathbf{p} = (x,y)^T$ and velocities $\mathbf{v} = (\dot{x},\dot{y})^T$ can be used. The direction of motion is used as orientation of the vessel θ. This orientation can be different to the true heading. However, for this work the assumption is that the heading corresponds with the direction of motion. Based on a starting state $\mathbf{s}_0$ the position of a vessel can be calculated at any time t with the system equation. Using the time different Δt relating to the starting time $\Delta t = t - t_0$ the system equation for the constant velocity model is given by:

$$s_{cv}(t) = \begin{pmatrix} x(t) \\ y(t) \\ \theta(t) \\ \dot{x}(t) \\ \dot{y}(t) \end{pmatrix} = \begin{pmatrix} x_0 + \dot{x} \cdot \Delta t \\ y_0 + \dot{y} \cdot \Delta t \\ \arctan\left(\frac{\dot{y}_0}{\dot{x}_0}\right) \\ \dot{x}_0 \\ \dot{y}_0 \end{pmatrix} \tag{5}$$

3.2 *Constant Turn Rate and Velocity (CTRV)*

The circular motion model is defied by a non zero turning rate $\omega = \theta$ and the constant track speed v. If turning rate is zero the CV model is used. The tangent velocity vector $v = (vx,vy)^T$ is defined by:

$$v(t) = \begin{pmatrix} vx(t) \\ vy(t) \end{pmatrix} = \begin{pmatrix} v \cdot \cos(\theta(t)) \\ v \cdot \sin(\theta(t)) \end{pmatrix} \tag{6}$$

The radius vector $\mathbf{r}(t)$ pointing from a vessel position $\mathbf{p}(t)$ to the centre of the circle is calculated using this vector $\mathbf{v}(t)$ and the turning rate:

$$r(t) = \frac{v(t)}{\omega}, \quad \omega \neq 0 \tag{7}$$

The position of the vessel $\mathbf{p}(t)$ on the circular path is calculated using the radius difference vector $\Delta\mathbf{r} = \mathbf{r}(t) - \mathbf{r}_0$. This radius difference vector $\Delta\mathbf{r}$ points from the starting position to the current position:

$$p(t) = p_0 + \Delta r \tag{8}$$

Applying equation 7 to equation 8 leads to:

$$p(t) = p_0 + \frac{v(t) - v_0}{\omega}, \ \omega \neq 0 \tag{9}$$

Using equation 9 for the vessel postion leads to the system equation for the CTRV model:

$$\mathbf{s}_{CTRV}(t) = \begin{pmatrix} x(t) \\ y(t) \\ \theta(t) \\ v(t) \\ \omega(t) \end{pmatrix} = \begin{pmatrix} x_0 + \frac{v\cdot(cos(\theta(t)) - cos(\theta_0))}{\omega} \\ y_0 + \frac{v\cdot(sin(\theta(t)) - sin(\theta_0))}{\omega} \\ \theta_0 + \omega\Delta t \\ v \\ \omega \end{pmatrix}, \omega \neq 0 \tag{10}$$

3.3 *Trajectory Following*

To estimate the position of a vessel traveling along a trajectory the trajecotry definition from equation 4 is used. The orientation of the vessel at any time $\theta(t)$ can by calculated using the time derivatives at position $\mathbf{p}(t)$:

$$\theta(t) = arcan\left(\frac{\dot{y}(t)}{\dot{x}(t)}\right) \tag{11}$$

The curvature $\kappa(t)$ of the curve can be calculated using the first and second time derivative at $\mathbf{p}(t)$:

$$\kappa(t) = \frac{\dot{x}(t)\ddot{y}(t) - \ddot{x}(t)\dot{y}(t)}{\left(\dot{x}(t)^2 + \dot{y}(t)^2\right)^{\frac{3}{2}}} \tag{11}$$

For this motion model the turning rate $\omega(t)$ changes over time and is calculated with the product of the track speed v and the curvature $\kappa(t)$:

$$\omega(t) = \kappa(t)\cdot v \tag{12}$$

Using equation (5), (11) and (13) leads to the system equation for a vessel moving along a trajectory defined as a sequence of Bézier curves:

$$\mathbf{s}_{TF}(t) = \begin{pmatrix} x(t) \\ y(t) \\ \theta(t) \\ v(t) \\ \omega(t) \end{pmatrix} = \begin{pmatrix} \sum_{i=0}^{n} b_{i,n}(c(t))P_{i_x,j} \\ \sum_{i=0}^{n} b_{i,n}(c(t))P_{i_y,j} \\ arctan\left(\frac{\dot{y}(t)}{\dot{x}(t)}\right) \\ v \\ \kappa(t)\cdot v \end{pmatrix}, \begin{matrix} t \in [T_{j-1}, T_j] \\ , 0 < j < k \end{matrix} \tag{13}$$

3.4 *Obstacle Handling with Safety Distance*

The information about the future motion, estimated either by CV model, CTRV model or the trajectory following method, is stored in a three dimensional grid with x,y and time. This grid is further called obstacle grid. All grid cells within a safety distance to the trajectories are marked as occupied and consequently prohibited for the own vessel. For the safety distance a Davis domain (Davis et al. 1982) scaled by the ship length is used to obtain a nautical behaviour taking into account the COLREGs.

4 TRAJECTORY GENERATION

For trajectory generation we use a discrete space as introduced above. Therefore, the own trajectory is mapped to the grid and compared with the obstacle grid. If a cell of the own path is marked as occupied by another vessel in the obstacle grid a potential collision is detected and an adopted A* algorithm is triggered to find a new set of waypoints for an avoidance trajectory. To guarantee the feasibility of moving along this waypoints an approach taking into account the kinematic constraints of a vessel as presented by Blaich et al. (2012a) is used. This approach only uses cells, reachable by the own vessel considering the heading in the current cell and the turning circle of the vessel. This constraint cell-neighbourhood is named as *T-Neighbourhood* because the reachable cells can be modelled as a T-shaped geometry. Because planning takes place only in a local area the waypoints of the avoidance trajectory are a subset of a larger global trajectory. A tangential connection between the avoidance trajectory and the global trajectory is used to reach a smooth transition. As presented by Blaich et al. (2012a) two virtual obstacles, representing the turning circle of the vessel, are used as so called *connection-funnel*. As search algorithm an A* algorithm using the T-Neighbourhood as presented by Blaich et al. (2012b) is applied and modified for the n-vessel scenarios with trajectory negotiation.

4.1 *Specialised A* for negotiated trajectories*

The A* search is applied to a grid. Each cell $\mathbf{c}$ of this grid represents the position of a vessel reaching this cell within a minimum of time. Additionally the orientation of the vessel, reaching this cell, is stored. Thus, the grid is 2.5D with $\mathbf{c} = (x,y,\theta)^T$ storing to each position one orientation. Suppose that the vessel moves on the grid it takes discrete steps k from cell $\mathbf{c}_k$ to cell $\mathbf{c}_{k+1}$. This motion is defined by an action u. The set of possible actions applicable from cell $\mathbf{c}_k$ is called action space $\mathbf{U}(\mathbf{c}_k)$. The T-Neighbourhood limits action space to a straight-line motion, a left and a right turn. The goal of the search algorithm is to find a sequence of reachable and collision free cells from a start cell $\mathbf{c}_0$ corresponding to position $\mathbf{p}_0$ to a goal cell $\mathbf{c}_g$ with minimum costs. As cost function the A*-algorithm combines a cost-to-come function $g(\mathbf{c}_k,u)$ with a heuristic cost-to-go function $h(\mathbf{c}_k)$ to estimate the total cost $f(\mathbf{c}_k,u)$ to reach a grid cell applying action u:

$$f(c_k) = g(c_k,u) + h(c_k) \tag{14}$$

For the cost-to-come function $g(\mathbf{c}_k,u)$, a simplified version without turn penalties but considering the covered distances $d(\mathbf{c}_k,u)$ for reaching cell $\mathbf{c}_k$ by applying action u to $\mathbf{c}_{k-1}$ is used. Additionally, the distance to the orginal trajectory d($\mathbf{c}_k$,$\mathbf{R}^0$) is used also. This leads to the cost-to-come function

$$g(c_k, u) = g(c_{k-1}) + \delta \cdot d(c_k, u) + \tau \cdot d(c_k, R^0) \qquad (15)$$

with δ and τ as scaling factors. For the distance function $d(\mathbf{c}_k,u)$ to the previous cell the Euclidian distance between the cells $\mathbf{c}_{k-1}$ and $\mathbf{c}_k$ is used. To calculate the distance between $\mathbf{c}_k$ and $\mathbf{R}^0$ an approximation of the area between $\mathbf{c}_{k-1}$ and $\mathbf{c}_{k-1}$ and the closest points to the original trajectory is used as trajectory distance function $d(\mathbf{c}_k,\mathbf{R}^0)$. For the approximation the average distance between $\mathbf{c}_{k-1}$ and $\mathbf{c}_{k-1}$ to $\mathbf{R}^0$ multiplied by the Euclidian distance between the two points on $\mathbf{R}^0$ is used. An illustration of this approximation is shown in Figure 2.

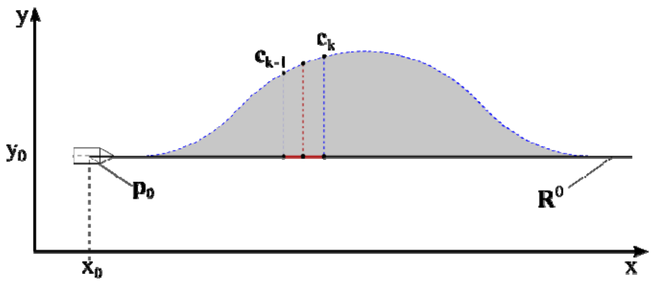

Figure 2. Distance between avoidance trajectory and original trajectory.

To use the area instead of the distance between $\mathbf{c}_k$ and $\mathbf{R}^0$ has the advantage that the distance $D(\mathbf{R}^*,\mathbf{R}^0)$ between the resulting trajectory $\mathbf{R}^*$ and the original trajectory $\mathbf{R}^0$ is the sum up of the distances $d(\mathbf{c}_k,\mathbf{R}^0)$ from the cell sequence estimated by the A* algorithm:

$$D(R^*, R^0) = \sum_{k=0}^{j} d(c_k, R^0) \qquad (17)$$

with trajectory $\mathbf{R}^*$ containing as waypoints all cells of the A* solution.

The complete cost of a trajectory is defined as the cost-to-come $g(\mathbf{c}_g,u)$ for the goal cell.

As heuristic function the Euclidian distance from cell $\mathbf{c}_k$ to the goal cell $\mathbf{c}_g$ is used.

The trajectory $\mathbf{R}^*$ estimated by the A*-algorithm contains as waypoints $\mathbf{W}^*$, a list of connected cells from the starting cell $\mathbf{c}_0$ to the goal cell $\mathbf{c}_g$. To achieve a trajectory $\mathbf{R}'$ as a sparse representation of $\mathbf{R}^*$, a Dougles-Peuker algorithm is used to eliminate needless waypoints using a certain threshold. This sparse trajectory $\mathbf{R}'$ makes it possible to exchange the complete trajectories between all agents during negotiation process.

5 TRAJECTORY NEGOTIATION

After the initial solution is found by the path-finding component, the locally planned avoidance trajectories are negotiated between all involved vessels. The initial, single trajectory for the own Vessel is planned, ignoring information about other vessel and used as an initial offer in the negotiation. This first trajectory is considered the desired trajectory of each vessel, if no other vessel would interfere. The value of other trajectories will be calculated in between this best solution and a hypothetical worst solution, which is usually unfeasible. In a number of negotiation rounds vessels broadcast their proposal for their trajectory in the set of all trajectories. At the end of the first round, intentions of all vessels are known, though they are very likely to conflict. In subsequent rounds, vessels discard sets with low value and re-plan their trajectory in sets with a high value. The new sets are exchanged as an offer while the global measure is designed to let the negotiation converge to an optimal, common solution. This procedure helps to compensate for incomplete information without the need to exchange state space information explicitly and balance the costs and benefits equally among all participants through the design of the trajectory-set selection process.

The following steps are our adoption of an algorithm of Steven P. Waslander (2007) who, in his thesis, designed distributed algorithms for agents which reach optimal solutions using concepts of game theory without the need to maintain all information locally at one central point. The *cooperative decentralised penalty method* is modified to work on the paths, already locally optimised by the path-finding component, to perform a global optimisation.

In our adopted algorithm, a number of agents $a \in A$ follow a number of trajectories $\mathbf{R}_a(\mathbf{W},v)$, consisting of waypoints $\mathbf{w}_0$ .. $\mathbf{w}_{j-1}$ where $\mathbf{w}_0$ is the starting position of the vessel which moves along that trajectory and $\mathbf{w}_{j-1}$ as its destination.

The original algorithm does not have a path-finding component which already minimises the distance from a desired optimal trajectory and the original trajectories are fully defined on discrete time steps. The path-finding component works on a smaller, minimal number of waypoints, as described in paragraph *4.1*. The agent cost function is therefore modified from the original approach to:

$$C_a(R_a) = \left\| R_a^d - R_a \right\| \qquad (19)$$

where R_a^d is the desired trajectory of agent a which is in our case a straight line from its position to its destination. Our decentralised augmented cost function contains no penalty function because the interconnected constraints, imposed by trajectories in the neighbourhood of the agents, are already fulfilled by the path-finding algorithm and trajectories which violate constraints cannot be generated. This leads to a decentralised augmented cost function:

$$C_a^{Aug}\left(R_a \mid \{R_i\}_a\right) = \beta_a\left(-\log\left(d_a - C_a(R_a)\right)\right) \qquad (20)$$

The function is defined on all neighbouring vessels $i \in A$ where $i \neq a$ which in our first approach includes all other vessel. The notation $\{R_i\}_j$ refers to the fixed knowledge of the agent of the trajectories of other agents. The disagreement point d_a, a point which represents the costs of the worst case solution, if the agents do not reach an agreement, is in our collision avoidance case defined as an usually unfeasible solution, which maximises the agent cost function. This results in a trajectory at the fringes of the state space where the distance between the ideal straight line trajectory is at its maximum. We use the same penalty parameter $\beta \geq 0$ as Inalhan et al. (2002) to ensure convergence of the cost function.

The decentralised optimisation problem is now to minimise equation (20) while we reduce β_i in order to reach the Nash Bargaining Solution. For an explanation of the cost decomposition of the Nash Bargaining Cost function from the central to the decentralised case the reader is referred to Waslander (2007). The detailed adopted algorithm, applied to our problem is as follows:

1 The first local trajectory of the own vessel is planned, using the A* component regardless of the other vessel's trajectories.
2 The waypoints of single trajectories from each agent are broadcasted, after which each agent has |a|-Sets of single trajectories, one for every agent and itself.
3 All sets of single trajectories are merged to one set R_a, with one trajectory for each agent. Also an initial value β_0 is chosen.
4 The disagreement point da is determined by maximising the agent cost function as described earlier.
5 Each agent performs a new search for a feasible trajectory for itself while keeping the others fixed. The dynamics of the system are taken care of by the path finding component. The search minimises the agent cost function and consequently also the augmented cost function. The new trajectory for the own agent merged with the other trajectories of set R_a^0 form the first full solution R_a
6 The full set R_a^1 is broadcasted to all other agents in the neighbourhood. After this step each agent has one full solution from each other agent.
7 In a while loop the following steps are repeated until the new planned trajectory at step t is less different from the previously planned at step t–1 by a defined ε.
8 The agents search in each received trajectory set a new own trajectory, using the path finding component, while keeping the other trajectories fixed. At the end of this step the augmented cost function gives an evaluation for each received solution set.
9 The agent select its preferred solution set R_a^t from all sets calculated which is the set with the lowest augmented cost.
10 The preferred set R_a is broadcasted to all agents in its neighbourhood and a new β_a chosen.
11 The while loop begins if the condition still holds.

In the domain it is important to improve best practises and collision avoidance manoeuvres and not interfere with operations in a way that application of the approach supersedes planned situation handling by legislative institutions. At the same time the approach should be equally beneficial to all vessels. Bargaining as a global decentralised search for a global Nash Solution is used to maximise the benefits for each vessel in an equal manner, in this case shorter trajectories. This is considered an additional incentive to use the system apart from safety concerns and similar approaches were used successfully in many different domains in the past as can be seen i.e. in Muthoo (1999). On a ship's bridge our approach can be used integrated in an ECDIS system as an expert system for mariners or it could be implemented in the ship`s system for autonomic collision avoidance.

6 EVALUATION

The procedure is currently evaluated in a simulation environment, implemented by the HTWG-Konstanz. In the situation, depicted in figure 5, two simulated vessels start approximately 600 meters apart with speed and heading chosen to lead to a crossing situation and, if not handled, to a collision in about 5 minutes. However, the system anticipates the collision and suggests a small detour for both vessels which, due to COLREGs, could not be handled that way once the vessel are in the crossing situation, but which is a more beneficial solution for both. Figure 5 shows an ongoing optimisation with the trajectories still being optimised and the desired trajectory of the vessel in the lower left corner, as a straight line.

In the simulation the convergence towards a global optimum could be observed. Since the agents exchange each set of trajectories in the simulation in several rounds, it could be observed that after the first trivial solution of straight lines, the path planning component planned a suboptimal solution in the sense that each agent tried to avoid the collision in the assumption that the other agents would not change its course. In the consequent rounds, the trajectories changed gradually towards the optimal solution of a straight line while minimising the augmented cost function and the overall trajectory length, as shown in figure 4. It can also be shown that the negotiations converge after 20-30 rounds towards an acceptable solution, and alternate between good solutions after 5-10 rounds.

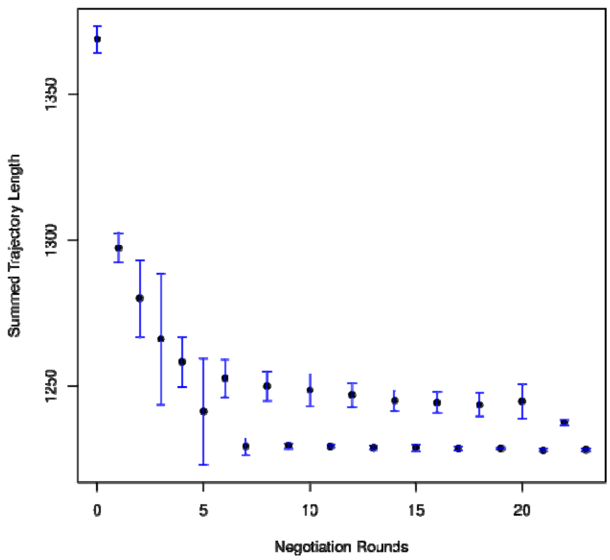

Figure 4. Average summed trajectory length in meters for all vessels in each negotiation round

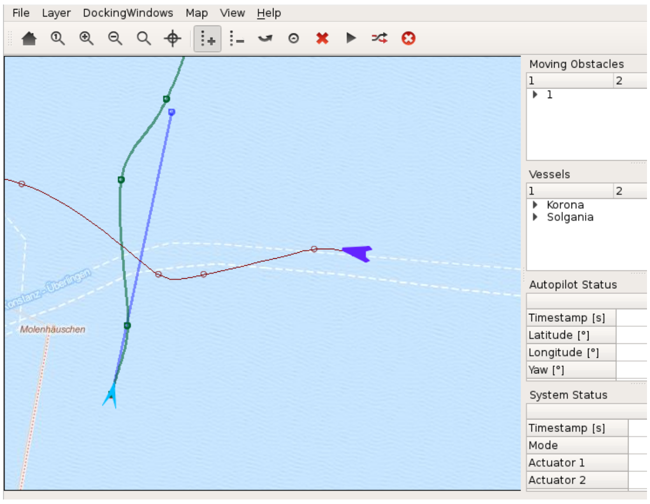

Figure 5. Two vessel just before a crossing situation.

The stability of the solutions found varied in the beginning because, in contrast to the constraint penalty and trajectory distance function used by Waslander (2007), our path finding led a to more complicated convergence behaviour. Towards a Pareto optimal solution, each vessel accepted solutions within a strictly optimising margin, however within this margin solutions were alternating between favourable solutions for each vessel.

Collision free trajectories could be found in most of the trials after the first or second round, which were considerable detours but safe. The approach is already tested on a test bed on Lake Konstanz using pleasure crafts. First results are in line with the simulation outcomes.

7 CONCLUSION

The procedure applies the negotiation schema from Waslander (2007) in the maritime domain and utilises a path-finding component to obey COLREGs and include the dynamic model of the ship. The negotiation converges towards a global optimum by using local optimisation and exchanging only trajectories. This leads to a collision free sub-optimal solution for all participating vessels after just a few negotiation rounds.

After first promising results from the simulation the next step is to verify the results in detail in the test beds on Lake Konstanz and in the eMIR test bed at the German Bight for merchant vessels and measure the impact of currents and changing situations on the negotiation progress.

REFERENCES

Blaich, M., Rosenfelder M., Schuster M., Bittel, O. & Reuter J. 2012a. Extended Grid Based Collision Avoidance Considering COLREGs for Vessels. In Proc. of the *9th IFAC Conference on Manoeuvring and Control of Marine Craft (MCMC)*

Blaich, M., Rosenfelder M., Schuster M., Bittel, O. & Reuter J. 2012b. Fast Grid Based Collision Avoidance for Vessels using A * Search Algorithm. In Proc. of the *17th International Conference on Methods and Models in Automation and Robotics (MMAR)*

Davis, P. V., Dove, M. J. & Stockel, C. T. 1982. A computer simulation of multi-ship encounters. *Journal of Navigation* 35(2):347-352

Heiskanen, P. 1999. Decentralized method for computing Pareto solutions in multiparty negotiations. *European Journal of Operational Research* 117(3), 578-590

Inalhan, G., Stipanovic, D. M., & Tomlin, C. J. 2002. Decentralized optimization, with application to multiple aircraft coordination. *Proceedings of the 41st IEEE Conference on Decision and Control* 1147-1155

Li, X. & Jilkov, V. 2003. Survey of maneuvering target tracking. Part I: Dynamic models. In IEEE Transactions on, *Aerospace and Electronic Systems* 39(4)

Mu, L., Kumar, R., & Prinz, A. 2011. An integrated wireless communication architecture for maritime sector. Multiple Access Communications (pp. 193-205). Springer Berlin Heidelberg

Muthoo, A. 1999. *Bargaining theory with applications*. Cambridge University Press

Schuster, M., Blaich, M. & Reuter J. 2014. Collision Avoidance for Vessels using a Low-Cost Radar Sensor. In Proc. of the *19th IFAC World Congress*

Smierzchalski, R. 1999. Evolutionary trajectory planning of ships in navigation traffic areas. *Journal of Marine Science and Technology* 4(1): 1-6

Szlapczynski, R. 2011. Evolutionary Sets Of Safe Ship Trajectories: A New Approach To Collision Avoidance. *Journal of Navigation* 64(1): 169-181

Szlapczynski, R. 2012. Evolutionary approach to ship ' s trajectory planning within Traffic Separation Schemes. *Polish Maritime Research* 19(1): 11-20

Szlapczynski, R. 2013. Evolutionary Sets of Safe Ship Trajectories Within Traffic Separation Schemes. *Journal of Navigation* 66(1): 65-81

Szlapczynski, R. 2015 Evolutionary Planning of Safe Ship Tracks in Restricted Visibility. *Journal of Navigation* 68(1): 39-51

Szlapczynski, R. & Szlapczynska J. 2012a. On evolutionary computing in multi-ship trajectory planning. *Applied Intelligence* 37(2): 155-174

Szlapczynski, R. & Szlapczynska J. 2012b. Evolutionary Sets of Safe Ship Trajectories: Evaluation of Individuals. *International Journal on Marine Navigation and Safety of Sea Transportation (TansNav)* 6(5): 345-353

Waslander, S. L. 2007. Multi-agent systems design for aerospace applications (*Doctoral dissertation*, Stanford University)