Parallel Automaton Representation of Marine Crafts' COLREGs-based Manoeuvering Behaviours

Papageorgiou, Dimitrios; Blanke, Mogens; Lützen, Marie; Bennedsen, Mette; Mogensen, John; Hansen, Søren

Published in:
IFAC-PapersOnLine

Link to article, DOI:
10.1016/j.ifacol.2019.12.291

Publication date:
2019

Document Version
Early version, also known as pre-print

Citation (APA):
Papageorgiou, D., Blanke, M., Lützen, M., Bennedsen, M., Mogensen, J., & Hansen, S. (2019). Parallel Automaton Representation of Marine Crafts' COLREGs-based Manoeuvering Behaviours. *IFAC-PapersOnLine*, *52*(21), 103-110. https://doi.org/10.1016/j.ifacol.2019.12.291

Parallel Automaton Representation of Marine Crafts' COLREGs-based Manoeuvering Behaviours

Dimitrios Papageorgiou * **Mogens Blanke** * **Marie Lützen** **
Mette Bennedsen ** **John Mogensen** ** **Søren Hansen** *

* *Department of Electrical Engineering, Technical University of Denmark, 2800 Lyngby, Denmark, (e-mail: dimpa,mb,soha@elektro.dtu.dk).*
** *Svendborg International Maritime Academy, 5700 Svendborg, Denmark, (e-mail: mal,jmo,mrb@simac.dk).*

Abstract: With international rules of navigation, the IMO COLREGS, describing the regulatory behaviours of marine vessels relative to each other, correct interpretation of situations is instrumental to the successful navigation at sea. This becomes even more crucial when temporal unattended bridge or fully unmanned navigation is aimed at. Based on a breakdown of COLREG rules, this paper presents a framework for representation of manoeuvering behaviours, that are expected when all vessels obey the rules. Our analysis is based on discrete-event systems theory and the proposed framework consists of sets of finite automata, segregating situation assessment from decision making. An intermediate supervisory layer coordinates the communication of these automata modules. The framework is tested in simulation environment using a realistic scenario.

Keywords: Autonomous vessels, situation assessment, discrete-event systems, finite-state automata, COLREGs representation, collision avoidance.

1. INTRODUCTION

Correct situation assessment and interpretation of the International Regulations for Preventing Collisions at Sea (COLREGs) is instrumental to safe navigation when marine vessels pass each other at sea. Aiming at a temporarily unattended bridge, a computer needs to make situation assessment and interpretation with the same reliability and accuracy as an experienced navigator. The present COLREGs have been in effect since 1972 (IMO, 2019), and while large number of studies have focused on algorithms to calculate paths for collision avoidance, the situation assessment and interpretation parts have been assumed as implicit prerequisites in most studies. With the possibility of having several different objects, such as land structures, buoys and other vessels within the zone of attention, these tasks become rather complex, and the algorithms for autonomous decision support need to be traceable to COLREGs, have no situations where deadlocks could occur and be provable correct. This paper employs Discrete-Event Systems (DES) theory to obtain these features of a generic representation.

Several algorithms for decision support in marine applications have been proposed in the literature. The authors in (Szlapczynski and Krata, 2018) presented a method for determining and visualizing ship collision avoidance information. The method employed evaluation of high-risk sectors based on calculation of the Closest Point of Approach (CPA). The focus was on decision support in the case of severe weather conditions. The CAPatternMinner framework for collision avoidance was presented in (Lei et al., 2018), where collision hazard detection was based on Automatic Identification System (AIS) data pattern analysis. COLREGs-compliant safety functions for collision risk reduction were presented in (Poo Arguelles et al., 2019). The implementation was based on a PLC/AIS communication bus framework. A proximity stage discrimination model for assessing Collision Risk (CR) during ship encounter was proposed in (Dinh and Im, 2017). The value of CR was calculated by a fuzzy set with the CPA and the Time for Closest Point of Approach (TCPA) being its inputs and was used to generate new waypoints for own ship. DES formalization for decision making support has been researched in several studies focusing on collision avoidance of ground and aerial vehicles. A theoretical framework for collision avoidance in vehicular intersections was presented in (Dallal et al., 2013), which introduced supervisory control in networks of controlled and uncontrolled vehicles. A controller for avoiding collisions in automated guided vehicle systems based on labelled Petri nets was developed in (Wan et al., 2018). Coloured Petri net formalism was used in (Tang et al., 2015) for preventing collision of airplanes. The framework was taking into account different pilot response times. A simplified DES-logic was used in (Moreno-Salinas et al., 2018) for formation control of surface marine vehicles. World automata were used in (Marinica et al., 2012) for collision avoidance in a group of aerial vehicles. The proposed framework comprised hybrid input/output automata for each vehicle, enriched with "world variables" that coordinated the parallel composition of these automata (Capiluppi and Segala, 2013).

Although the problem of collision avoidance has been adequately handled by several DES-based algorithms in the previous works, the explicit representation of the relevant COLREGs for autonomous decision support in surface vessels has sparsely been addressed. This paper presents a three-layer framework for autonomous decision support for navigation based on DES theory. Specifically, two separate Deterministic Finite-State Automata (DFA) are used for diagnosis of current situation and decision making. An additional supervisory module acts as an intermediate layer that manages the interaction between the situation-assessment and the decision-making automata. The advantage of this modular structure pertains not only to implementation simplicity but also to the fact that it allows independent runtime operation of the situation-assessment (awareness) and decision-making (navigation-manoeuvering) loops. Moreover, alteration of each of the modules, such as adding more diagnostic steps or more actions, can be done independently from each other. This, in turn, facilitates greater design flexibility and easier maintenance of the autonomous decision support system. One last feature that stems from the DES formulation relates to health monitoring of several of the ship components and equipment, since fault detection is possible by means of modes-events tracking (Blanke et al., 2015).

The remaining of the paper is structured as follows: Section 2 gives an overview of an autonomous sea-faring system for marine surface vessels and describes a typical situation-assessment/decision-making cycle. Section 3 details the DES-based framework for decision support. The applicability of the proposed approach is demonstrated in a simulation of a real-case scenario in Section 4, where comments are made on the results. Finally, conclusions are drawn in Section 5 and elements of future work are presented.

2. SYSTEM DESCRIPTION

Marine surface vessels operation systems consist of three basic components. The first is the sensing component, which includes all the diagnostic subsystems and equipment, e.g. sensors and sensor fusion modules. The second component relates to the ship actuation and includes the actuators of the vessel, such as propellers, rudders etc., as well as the associated control systems. Finally, the third component consists of all the coordination and supervisory modules that facilitate the connection between the sensing and actuation modules. This essentially corresponds to the necessary "logic" that interprets the information provided by the sensors and makes decisions on the next course of actions.

The information collected by the ship sensors, such as radar, cameras, lidars, Electronic Chart Display and Information System (ECDIS), AIS, Global Positioning System (GPS), Inertial Navigation System (INS), is consolidated via sensor fusion and passed on to both the ship motion control systems and the situation-assessment module. The latter evaluates the risk of potential hazards. Such assessment includes conclusions on the relative position of the own ship with respect to other vessels or land, as well as predicted behaviours of other vessels. Finally, the situation-assessment block provides an awareness input to the supervisory module, which evaluates the situation based on the own ship's prescribed route and gives the appropriate commands to the guidance, navigation and motion control systems of the ship. The general operation scheme of a ship is shown in Figure 1.

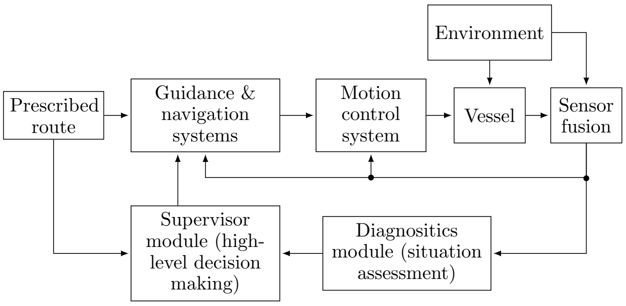

Fig. 1. Block diagram of situation awareness and navigation. Sensor information is also directly passed to the guidance and navigation systems for facilitating feedback control algorithms.

A typical chain of situation assessment and decision making functions during navigation is shown in the schematics of Figures 2 and 4. As soon as the radar/cameras detect an object, the CPA, i.e. the shortest distance of the own ship to the object over all time (Hansen et al., 2013), is calculated and compared to an awareness distance limit d_{aw}. If this limit is violated, then the TCPA is estimated and compared to the equivalent awareness time limit t_{aw}. In the case of this limit also being violated, further examination of the detected object is needed. Specifically, the situation-assessment loop continues as follows:

First, the ECDIS readings are compared to those of the sensor fusion module (radar, cameras etc.) to determine if the detected object is land, structure, buoy or a moving vessel. In the case of land, structure or buoy, the shape of the object from the ECDIS is compared to that from the cameras and if there are substantial discrepancies, an alarms is triggered. If on the other hand the type and shape of the object is confirmed by the ECDIS, then both the own ship position and that of the detected object are confirmed by calculating a *certainty factor*. This factor is a metric of how reliable the position estimations are, given certain conditions, such as weather, traffic, available light, visibility etc. If the certainty factor is below a limit, then an alarm is triggered, otherwise the estimated positions are used to determine whether alteration of the own ship's course is necessary or not. If this is the case, then an appropriate manoeuver is planned and commanded to the autopilot. After the manoeuver is performed, the ship returns to its normal course.

If the detected object is a moving vessel, then the relative bearing between it and the own ship is calculated as shown in Figure 3. If the other vessel is performing an overtaking manoeuvre, the TCPA is compared to the Time to Next Waypoint (TTW) of the own ship. If the TCPA is smaller than TTW and the CPA is smaller than a pre-specified distance d_{act} (also called *action limit*), then there is a high risk of collision and an alarm is raised. A similar hazard arises when TCPA is larger than the TTW but the next waypoint of the own ship is towards the moving vessel, in which case the same alarm is triggered. If none of these cases hold, then the ship continues on its predefined course. A similar approach is made if the moving vessel is on the starboard side of the own ship. The difference with the previous case is that there is an additional option of taking a special action when both CPA and TCPA limits are violated, instead of triggering the alarm. This can be seen in Figure 4. If the moving vessel is on the port side of the own ship, then depending on whether it is a motor vessel or not, the same actions are followed as in the overtaking or starboard cases, respectively. Finally, if the relative bearing cannot be determined, an alarm is raised.

Remark 1. There are several types of alarms that depend on the situation-assessment chain that has led their triggering. For instance, inability to determine the relative bearing between own ship and the other vessel implies that the detected object is too fast to estimate its position, whereas very low position certainty factor leads to an alarm relative to the sensors reliability. Another example is the case where TCPA is larger than TTW but the next waypoint of the own ship is on the same side as the detected vessel (boxes 9 and 10 in Figure 2). Here the alarm may suggest an entirely different approach, such as keeping the same course and speed, irrespectively of the next waypoint. Finally, if an alarm is raised due to discrepancy between the sensor fusion and the ECDIS, then a different diagnosis is triggered to determine whether the expected object is missing, has different shape and colour etc. The actions taken after an alarm is triggered also differ depending on the alarm type. They can range from performing more complicated manoeuvres to immobilizing the ship. Analytical description of all possible alarm types is beyond the scope of this paper.

Remark 2. The relative position between the own ship and the detected vessel is classified in the three categories mentioned earlier in this section as follows: If the other vessel has relative bearing $-112.5^o \leq \beta \leq -3.5^o$, it is on the port side. If $-3.5^o \leq \beta \leq 112.5^o$, it is on the starboard side. Otherwise is overtaking as shown in figure 3. Under certain visibility conditions, it may be hard to determine the border line between port and starboard configuration and therefore, it is common to allow for a small uncertainty region $-3.5^o \leq \beta \leq 0$, in which it is considered that the other vessel is on the starboard side (Cockcroft and Lameijer, 2011; Lyu and Yin, 2019).

Remark 3. Similarly to the case of the alarms, different special actions can be taken, depending on several factors, such as the area of sailing, the traffic, the manoeuvrability of the own ship, the weather conditions, the condition of the sensing equipment etc.. Such special actions can range from simple speed reduction to turning starboard or even performing more complicated manoeuvres. A detailed listing of all possible special actions is outside of the scope of this study and will be omitted from this paper.

3. DECISION SUPPORT FRAMEWORK

The clear segregation of a navigator's activities, as these were described in Section 2, into situation assessment and decision making motivates the use of a modular structure for sea-faring autonomy that facilitates collision avoidance. DES theory and specifically DFA constitute a natural framework for developing both diagnosis and decision-making solutions for autonomous systems due to the event-driven dynamics of such processes. The proposed framework comprises two DFA, one for situation assessment and one for decision-making and a supervisory module that coordinates the two automata. One advantage that motivated the selected framework choice was also the ability to detect deadlocks based on the connectivity properties of the DFA graphs, as this will be demonstrated later in this section. Both the situation-assessment and the decision-making DFA are detailed below. Elements of DES theory are provided in Appendix A.

3.1 Situation-assessment automaton

The information reported to the own ship supervisory module relates to the following statements:

r_0 : The sensor fusion (camera/radar) system detects an object.
r_1 : The detected object violates the awareness level in terms of CPA (CPA$< d_{aw}$).
r_2 : The detected object violates the awareness level in terms of TCPA (TCPA$< t_{aw}$).
r_3 : The ECDIS reports that the object is land.
r_4 : The ECDIS reports that the object is buoy.
r_5 : The geometry of the detected object matches the ECDIS description.
r_6 : The detected moving object is overtaking.
r_7 : The detected moving object is on the port side of the own ship.
r_8 : The detected moving object is on the starboard side of the own ship.
r_9 : The detected moving object is a motor vessel.
r_{10} : The CPA is smaller than the action limits d_{act}.
r_{11} : The TCPA is smaller than the TTW.
r_{12} : The next course is on the same side as the detected moving object.
r_{13} : The certainty factor of sensor fusion description matching the ECDIS is above the predefined threshold.
r_{14} : There is conflict between the predefined route and the land/buoy position.
r_{15} : The diagnostic report has been passed to the supervisor.
r_{16} : The course adjustment is completed.
r_{17} : The special action is taken.

The discrete states $D_i : \bigcup_{i=1}^{16} \{D_i\} \triangleq \mathcal{D}$ concerning the situation assessment chain are:

D_1: The sensor fusion system (radar, camera, etc.) scans for objects (initial state).
D_2: The CPA is compared to r_{aw} (awareness distance limits).
D_3: The TCPA is compared to t_{aw} (awareness time limits).
D_4: The ECDIS readings are used to assess whether the detected object is land or construction.
D_5: The ECDIS readings are used to assess whether the detected object is buoy.

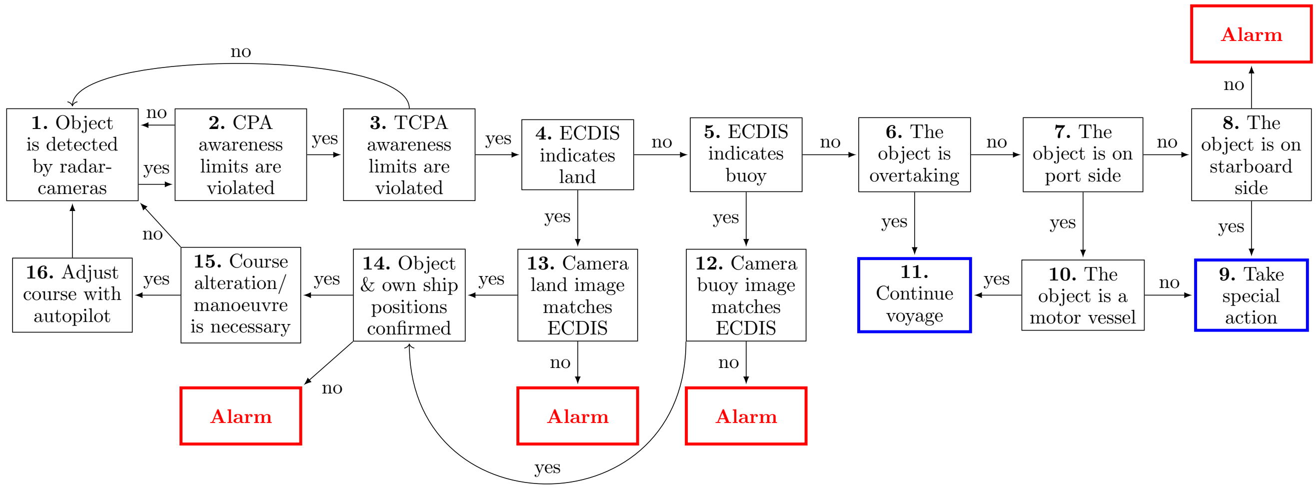

Fig. 2. Situation-assessment and decision-making loops based on COLREGs.

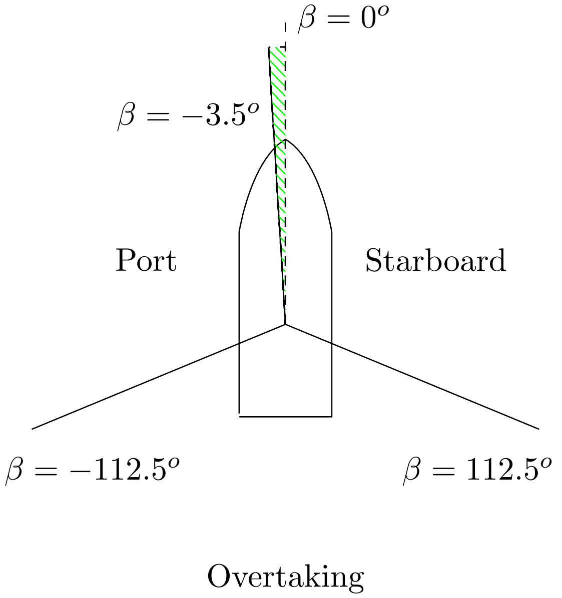

Fig. 3. Determination of relative position between own ship and detected moving object. The coloured sector corresponds to the area of the head-on uncertainty.

D_6: An assessment is made of whether the moving object is overtaking the own ship.
D_7: An assessment is made of whether the moving object is on the port side the of own ship.
D_8: An assessment is made of whether the moving object is on the starboard side of the own ship.
D_9: An assessment is made of whether the moving object is a motor vessel or not.
D_{10}: The TCPA is compared to the TTW.
D_{11}: The CPA is compared to the action limits d_{act}.
D_{12}: An assessment is made of whether the next course of the own ship is on the same side of the detected moving object.
D_{13}: The geometry of the detected object from the sensor fusion system is compared to the that coming from ECDIS.

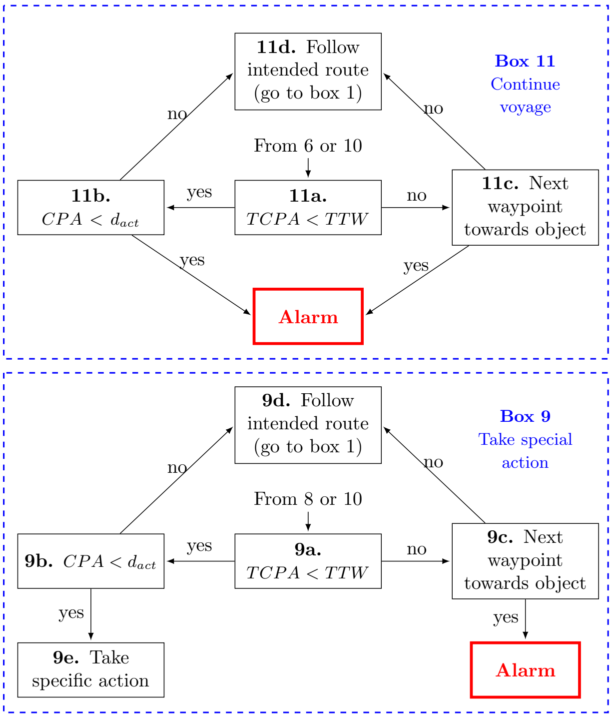

Fig. 4. Expanded blocks 9 and 11.

D_{14}: The position certainty factor is calculated and compared to the predefined threshold.
D_{15}: The current course and next waypoint of the own ship is compared to the position of the detected object.
D_{16}: The sequence of the reports is passed to the supervisor (marked state).

The event set E_D associated to the discrete states set $\mathcal{D}$ is defined as:

$$E_d = \bigcup_{i=0}^{16} \{r_i, \neg r_i\}.$$

The extended transition function $f_d : \mathcal{D} \times E_d^* \to \mathcal{D}$ is fully described by the state transition diagram in Figure 5. For instance $f_d(D_1, \neg r_0) = D_1$, $f_d(D_7, \neg r_7) = D_8$,

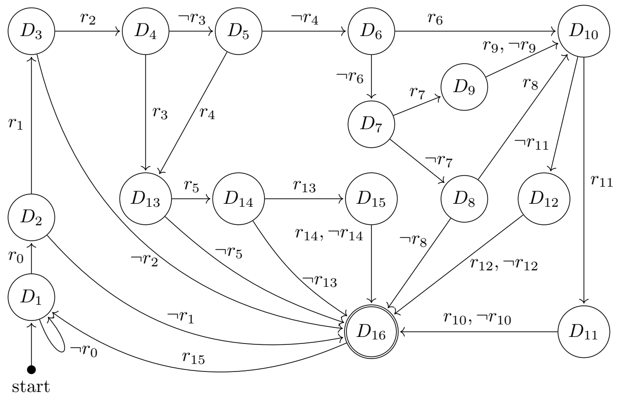

Fig. 5. State transition diagram of the situation-assessment automaton.

$f_d(D_2, r_1 r_2) = f_d(f_d(D_2, r_1), r_2) = f_d(D_3, r_2) = D_4$ and $f_d(D_{15}, \varepsilon) = D_{15}$, while for transitions not described in the diagram the function is undefined, e.g. $f_d(D_8, r_9)$ is not defined. Then the situation-assessment automaton G_d is defined as the five-tupple

$$G_d \triangleq (\mathcal{D}, E_d, f_d, D_1, D_{16}) \, . \tag{3.1}$$

The set of marked states for G_d is chosen to contain only the state D_{16} because it signifies the end of a full diagnosis cycle. The language generated by the G_d is defined as

$$\mathcal{L}(G_d) \triangleq \{s \in E_d^* \ : \ f_d(D_1, s) \text{ is defined}\} \tag{3.2}$$

and is basically the set of all strings s of events in E_d that starting from the initial sate D_1 cause a valid transition to any of the sates in $\mathcal{D}$, i.e an admissible directed path in the state transition diagram of Figure 5. The language $\mathcal{L}(G_d)$ is a powerful tool for fault detection, since invalid transitions, i.e. observed strings $s_f \notin \mathcal{L}(G_d)$, imply the occurrence of faults in the situation-assessment process. It should be noted that $\mathcal{L}(G_d)$ is by definition prefixed-closed since a string $s \in \mathcal{L}(G_d)$ only if the all of its prefixes correspond to possible paths in the situation-assessment automaton graph (Cassandras and Lafortune, 2008).

The language $\mathcal{L}_m(G_d)$ marked by G_d is defined as

$$\mathcal{L}_m(G_d) \triangleq \{s \in \mathcal{L}(G_d) \ : \ f_d(D_1, s) = D_{16}\} \ . \tag{3.3}$$

If L_d is the concatenated language $L_d \triangleq \{\varepsilon\}^* \bigcup \{\neg r_0\}^* L_1^{16}$, where L_1^{16} contains all the strings of length less or equal the cardinality of $E_d - \{\neg r_0\}$ that cause a transition from D_1 to D_{16}, then $\mathcal{L}_m(G_d)$ is the Kleene-Closure of L_d.

The situation-assessment automaton G_d is not blocking since from each mode D_i there exists a path in the transition diagram to d_{16}, i.e all its states are coaccessible to D_{16}.

3.2 Decision-making automaton

The discrete states $Q_i : \bigcup_{i=1}^{4} \{Q_i\} \triangleq \mathcal{Q}$ concerning the own ship operation are:

Q_1: The own ship moves on the predefined course.
Q_2: The own ship makes a course adjustment.
Q_3: A special action is taken.
Q_4: The alarm is raised.

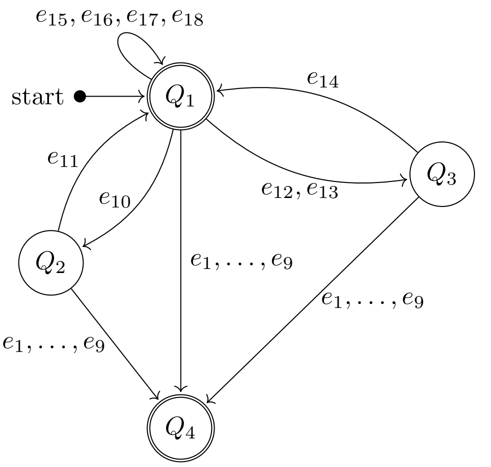

Fig. 6. State transition diagram of the decision-making automaton.

The event set E_q governing the transitions between the modes Q_i is defined as $E_q \triangleq \bigcup_{i=1}^{18} \{e_i\}$, where the events e_i are considered after the possible paths in the schematic of Figure 2. These events will be explicitly defined later in this section along with the description of the supervisory module.

Similarly to the case of G_d, the extended transition function $f_q : \mathcal{Q} \times E_q^* \to \mathcal{D}$ is fully described by the state transition diagram in Figure 6. For example $f_q(Q_1, e_{10}) = Q_2$, whereas $f_q(Q_3, e_{11})$ is undefined. The decision-making automaton G_q is defined as the five-tupple

$$G_q \triangleq (\mathcal{Q}, E_q, f_q, Q_1, \mathcal{Q}_m), \ \mathcal{Q}_m = \{Q_1, Q_4\}. \tag{3.4}$$

The language generated by the G_q is defined as

$$\mathcal{L}(G_q) \triangleq \left\{s \in E_q^* \ : \ f_q(Q_1, s) \text{ is defined}\right\} \ . \tag{3.5}$$

The language $\mathcal{L}_m(G_q)$ marked by G_q is defined as

$$\mathcal{L}_m(G_q) \triangleq \{s \in \mathcal{L}(G_q) \ : \ f_q(Q_1, s) \in \mathcal{Q}_m\} \ . \tag{3.6}$$

Similarly to G_d, the decision-making automaton G_q is not blocking since all its states are coaccessible to $\mathcal{Q}_m$.

3.3 Supervisory module

The supervisory module is an intermediate layer in the autonomous decision support framework that receives inputs from the situation-assessment automaton, as well as from the guidance and motion control systems of the ship. The input from G_d is an element of L_d, i.e. a series of events at the end of each diagnostics cycle. After processing the received inputs, the supervisory module outputs an event string $s \in L_d = e_i, \ i \in \{1, 2, 3, \ldots 18\}$ generated by the projection of the situation-assessment DFA event sequence onto the event set $E_d^- \triangleq E_d - \{r_{15}\}$ (basically by removing the last event r_{15}). This event string triggers a transition in the decision-making automaton described earlier in this section. The evaluation of each event e_i depends on the elements r_i received from the situation-assessment automaton. By denoting the logical OR operator with the "+" symbol, the events e_i are defined as follows:

$$
\begin{aligned}
e_1 &= r_0 r_1 r_2 (r_3 + \neg r_3 r_4) \neg r_5 \\
e_2 &= r_0 r_1 r_2 \neg r_3 \neg r_4 r_6 r_{11} r_{10} \\
e_3 &= r_0 r_1 r_2 \neg r_3 \neg r_4 r_6 \neg r_{11} r_{12} \\
e_4 &= r_0 r_1 r_2 \neg r_3 \neg r_4 r_7 r_9 r_{11} r_{10} \\
e_5 &= r_0 r_1 r_2 \neg r_3 \neg r_4 r_7 r_9 \neg r_{11} r_{12} \\
e_6 &= r_0 r_1 r_2 \neg r_3 \neg r_4 r_7 \neg r_9 \neg r_{11} r_{12} \\
e_7 &= r_0 r_1 r_2 \neg r_3 \neg r_4 r_8 \neg r_{11} r_{12} \\
e_8 &= r_0 r_1 r_2 \neg r_3 \neg r_4 \neg r_6 \neg r_7 \neg r_8 \\
e_9 &= r_0 r_1 r_2 (r_3 + \neg r_3 r_4) r_5 \neg r_{13} \\
e_{10} &= r_0 r_1 r_2 (r_3 + \neg r_3 r_4) r_5 r_{13} r_{14} \\
e_{11} &= r_{16} \\
e_{12} &= r_0 r_1 r_2 \neg r_3 \neg r_4 r_7 \neg r_9 r_{11} r_{10} \\
e_{13} &= r_0 r_1 r_2 \neg r_3 \neg r_4 r_8 r_{11} r_{10} \\
e_{14} &= r_{17} \\
e_{15} &= r_0 r_1 r_2 \neg r_3 \neg r_4 \left(r_6 + \neg r_6 r_7 (r_9 + \neg r_9)\right. \\
&\quad \left. + \neg r_6 \neg r_7 r_8\right) r_{11} \neg r_{10} \\
e_{16} &= r_0 r_1 r_2 \neg r_3 \neg r_4 \left(r_6 + \neg r_6 r_7 (r_9 + \neg r_9)\right. \\
&\quad \left. + \neg r_6 \neg r_7 r_8\right) \neg r_{11} \neg r_{12} \\
e_{17} &= r_0 \neg r_1 \\
e_{18} &= r_0 r_1 \neg r_2
\end{aligned}
$$

By inspection of the sequences in E_d and the definition of the values e_i, it is easy to see that except for e_{11} and e_{14}, the events in the output sequence are mutually exclusive. This means that there can not be ambiguity regarding the transitions in the decision-making automaton. This is not surprising since the definitions of the events e_i in the supervisory module were made in accordance to the possible outcomes of the situation-assessment/decision-making loops in Figure 2.

The situation-assessment and decision-making automata operate independently to each other and at different frequencies. A cycle in the diagram of G_d should be as fast as the sensor fusion modules allow, while the duration of a cycle in G_q depends on the specific situation. For instance, if nothing unexpected occurs during the trip of the own ship, the decision-making automaton will always be in Q_1.

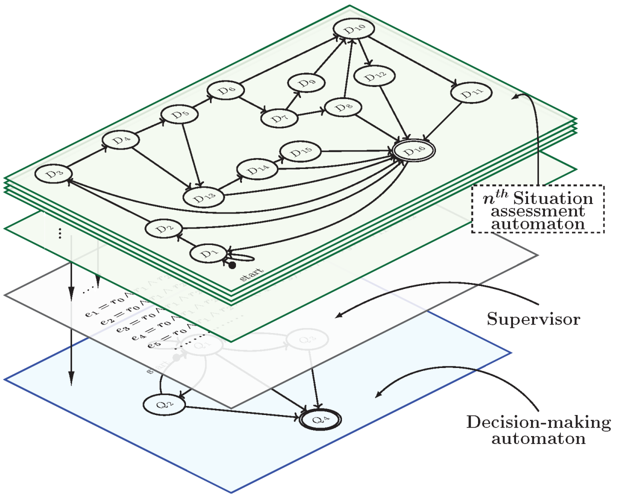

Fig. 7. State transition diagram of the parallel composition of decision-making and situation-assessment automata via the modular architecture and the supervisor layer.

In the case of multiple detected objects, separate instances of the situation-assessment automaton run in parallel and provide all information to the supervisory model, which employs either consensus or optimisation methods. This case is not considered in the present paper.

Remark 4. The DFA state transition diagrams were manually generated based on the possible diagnostic actions during situation assessment and on the transition rules that defined the supervisory module. As such, all possible transitions were included.

4. SIMULATION RESULTS

The autonomous decision support framework was implemented and tested in simulation environment using Matlab. The simulator included a kinematics model for the own ship and a second vessel as well as the three-layer proposed solution, without considering specific models for sensor fusion. The operation scenario is detailed in the following.

4.1 Simulation scenario

The simulation scenario considers a small power-driven own ship of 48m length, 6m breadth and 144m turning radius manoeuvrability, sailing on open waters with 16 knots on a course 57^o from north. The weather is clear with no rain and no depth or navigational restrictions, while the traffic congestion is low. At $t = 0$ an object is observed on the radar scans at distance 11.5 nautical miles. The detected object is a vessel moving with 12 knots at zero heading in 50^o bearing relatively to the own ship. The awareness and action limits are chosen as $d_{aw} = 2.7$ nautical miles, $d_{act} = 1.5$ nautical miles, $t_{aw} = 10$ minutes and $TTW = 6$ minutes. It is assumed that there are no discrepancies in the sensors or the ECDIS readings. The scenario was simulated for an hour (simulation time) with sampling time of 10sec.

4.2 Results

According to the diagram in Figure 2, at a given time when both the CPA and TCPA limits are violated and since the detected vessel is on the starboard side of the own vessel, a special action will have to be taken. According to the COLREGs rule 15 (see Appendix B), the action relates to the own ship giving way to the target vessel, which in this scenario is facilitated by a turn to starboard by 15^o. The event sequence that will be output by the situation-assessment automaton is

$$s = r_0 r_1 r_2 \neg r_3 \neg r_4 \neg r_6 \neg r_7 r_8 r_{11} r_{10} r_{15} \ .$$

Indeed, after approximately 45min, the own ship makes a starboard turn of 15^o to avoid risk of collision with the detected vessel. The modes and events sequences for both G_d and G_q are listed below for the time interval $43-47$min:

$$D_1 \to D_2 \to D_3 \to D_4 \to D_5 \to D_6 \to D_7 \to D_8 \to D_{10} \to D_{12} \to D_{16} \to D_1 \to D_2 \to D_3 \to D_4 \to D_5 \to D_6 \to D_7 \to D_8 \to D_{10} \to D_{11} \to D_{16} \ .$$

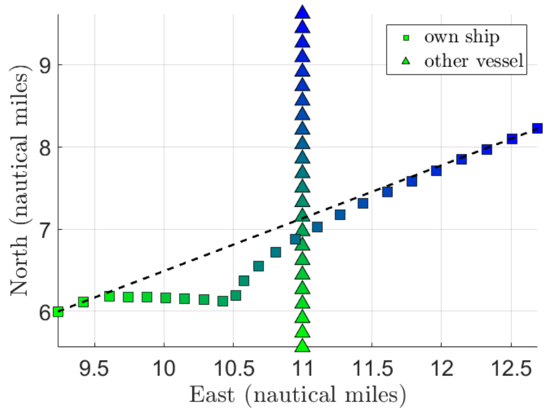

Fig. 8. Simulated courses of own ship and other vessel. The marker colours denote the evolution in time. Green colour corresponds to initial times, while blue to final times. It can be seen that the 15^o starboard turn of the own ship allowed the other vessel to pass from the intersection point earlier than the own ship.

$$\begin{aligned} e_a &= r_0 r_1 r_2 \neg r_3 \neg r_4 \neg r_6 \neg r_7 r_8 \neg r_{11} \neg r_{12} r_{15} \\ &\Rightarrow e_{16} = P_{E_d^-}(e_a) \text{ and } Q_1 \to Q_1 \ , \\ e_b &= r_0 r_1 r_2 \neg r_3 \neg r_4 \neg r_6 \neg r_7 r_8 r_{11} r_{10} r_{15} \\ &\Rightarrow e_{13} = P_{E_d^-}(e_b) \text{ and } Q_1 \to Q_3 \ . \end{aligned}$$

The parallel composition of G_d and G_q through the supervisory module can be written for these time interval as

$$\begin{aligned} &(D_1,Q_1) \xrightarrow{r_0} (D_2,Q_1) \xrightarrow{r_1} (D_3,Q_1) \xrightarrow{r_2} (D_4,Q_1) \xrightarrow{\neg r_3} \\ &(D_5,Q_1) \xrightarrow{\neg r_4} (D_6,Q_1) \xrightarrow{\neg r_6} (D_7,Q_1) \xrightarrow{\neg r_7} (D_8,Q_1) \xrightarrow{r_8} \\ &(D_{10},Q_1) \xrightarrow{\neg r_{11}} (D_{12},Q_1) \xrightarrow{\neg r_{12}} (D_{16},Q_1) \xrightarrow{r_{15}} (D_1,Q_1) \xrightarrow{r_0} \\ &(D_2,Q_1) \xrightarrow{r_1} (D_3,Q_1) \xrightarrow{r_2} (D_4,Q_1) \xrightarrow{\neg r_3} (D_5,Q_1) \xrightarrow{\neg r_4} \\ &(D_6,Q_1) \xrightarrow{\neg r_6} (D_7,Q_1) \xrightarrow{\neg r_7} (D_8,Q_1) \xrightarrow{r_8} (D_{10},Q_1) \xrightarrow{r_{11}} \\ &(D_{11},Q_1) \xrightarrow{r_{10}} (D_{16},Q_1) \xrightarrow{r_{15}} (D_1,Q_3) \ . \end{aligned}$$

Figure 8 shows the trajectories of both the own ship and the detected vessel, denoted by squares and triangles, respectively. The colour convention used is indicative of time, i.e. green corresponds to initial and blue to final times. As it can be seen, the starboard turn performed by the own ship allowed enough time for the other vessel to pass through the courses' intersection area in a safe distance to avoid collision risk. This is also shown in Figure 9, where the entire courses of the own ship and the detected vessel are plotted with (top) and without (bottom) the own ship special action taken.

5. CONCLUSIONS AND FUTURE WORK

A three-layer DES-based framework was presented in this paper for describing COLREGs-based manoeuvring behaviours of marine surface vessels. The proposed solution was based on a segregated representation of the diagnosis and decision-making functionalities that are taking place before, during and after various objects are detected on the ships's sensors. Two DFA were introduced modelling all the controls and actions advised by COLREGs. An additional supervisory module was used for coordination of the automata. The usability of this low-complexity framework was demonstrated in simulation environment in a scenario of a single vessel approaching the own ship with high risk of collision. Autonomous situation assessment and decision making were achieved through appropriate event monitoring and mode scheduling by the supervisory module. Future extensions of this work will include consideration of more complicated scenarios with more than one encountered objects, as well as the design of DES observers for state tracking.

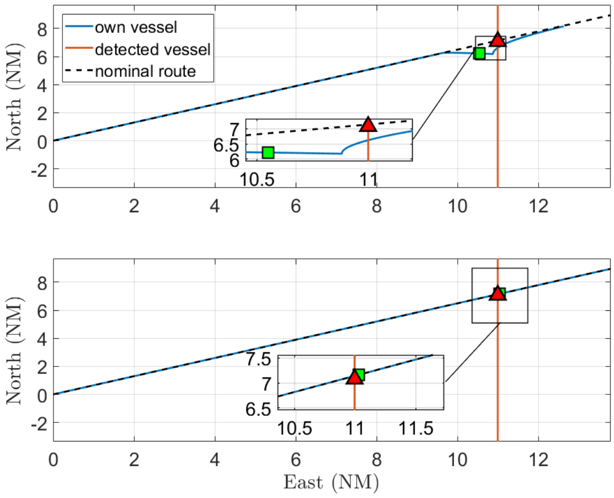

Fig. 9. Simulated courses of own ship and other vessel in the case of changed (top) and unchanged (bottom) course. The square and the triangle markers depict the own ship and the other vessel, respectively at the TCPA. The origin of the coordinate system is the initial position of the own ship.

ACKNOWLEDGEMENTS

This research is sponsored by the Danish Maritime Foundation, Orients Fund and LauritzenFonden through the ASAN project.

REFERENCES

Blanke, M., Kinnaert, M., Lunze, J., and Staroswiecki, M. (2015). *Diagnosis and Fault-tolerant Control, 3rd Edition*. Springer. doi:10.1007/978-3-662-47943-8.

Capiluppi, M. and Segala, R. (2013). World automata: A compositional approach to model implicit communication in hierarchical hybrid systems. *Electronic Proceedings in Theoretical Computer Science, Eptcs*, 124, 58–72. doi:10.4204/EPTCS.124.7.

Cassandras, C.G. and Lafortune, S. (2008). *Introduction to discrete event systems*. Springer US. doi:10.1007/978-0-387-68612-7.

Cockcroft, A. and Lameijer, J. (2011). *A Guide to the Collision Avoidance Rules*. Elsevier Science.

Dallal, E., Colombo, A., Del Vecchio, D., and Lafortune, S. (2013). Supervisory control for collision avoidance in vehicular networks using discrete event abstractions. *Proceedings of the American Control Conference*, 6580514, 4380–4386. doi:10.1109/ACC.2013.6580514.

Dinh, G.H. and Im, N.K. (2017). A study on the construction of stage discrimination model and consecutive waypoints generation method for ship's automatic avoiding action. *International Journal of Fuzzy Logic and Intelligent Systems*, 17(4), 294–306. doi:10.5391/IJFIS.2017.17.4.294.

Hansen, M.G., Jensen, T.K., Lehn-Schioler, T., Melchild, K., Rasmussen, F.M., and Ennemark, F. (2013). Empirical ship domain based on ais data. *Journal of Navigation*, 66(6), 931–940. doi:10.1017/S0373463313000489.

IMO (2019). *Convention on the International Regulations for Preventing Collisions at Sea, 1972 (COLREGs).* `http://http://www.imo.org/en/About/Conventions/ListOfConventions/Pages/COLREG.aspx` [Accessed: July 03].

Lei, P.R., Xiao, L.P., Wen, Y.T., and Peng, W.C. (2018). Capatternminer: Mining ship collision avoidance behavior from ais trajectory data. *International Conference on Information and Knowledge Management, Proceedings*, 1875–1878. doi:10.1145/3269206.3269221.

Lyu, H. and Yin, Y. (2019). Colregs-constrained real-time path planning for autonomous ships using modified artificial potential fields. *Journal of Navigation*, 72(3), 588–608. doi:10.1017/S0373463318000796.

Marinica, N.E., Capiluppi, M., Rogge, J.A., Segala, R., and Boel, R.K. (2012). Distributed collision avoidance for autonomous vehicles: World automata representation. *Ifac Proceedings Volumes (ifac-papersonline)*, 45(9), 216–221. doi:10.3182/20120606-3-NL-3011.00088.

Moreno-Salinas, D., Crasta, N., Pascoal, A.M., and Aranda, J. (2018). Formation control of surface marine vehicles for underwater target tracking using range information. *2018 13th Apca International Conference on Control and Soft Computing (controlo). Proceedings*, 201–6, 201–206. doi:10.1109/CONTROLO.2018.8514303.

Poo Arguelles, R., Garcia Maza, J.A., and Mateos Martin, F. (2019). Specification and design of safety functions for the prevention of ship-to-ship collisions on the high seas. *Journal of Navigation*, 72(1), 53–68. doi:10.1017/S0373463318000553.

Szlapczynski, R. and Krata, P. (2018). Determining and visualizing safe motion parameters of a ship navigating in severe weather conditions. *Ocean Engineering*, 158, 263–274. doi:10.1016/j.oceaneng.2018.03.092.

Tang, J., Piera, M.A., and Baruwa, O.T. (2015). A discrete-event modeling approach for the analysis of tcas-induced collisions with different pilot response times. *Proceedings of the Institution of Mechanical Engineers, Part G: Journal of Aerospace Engineering*, 229(13), 2416–2428. doi:10.1177/0954410015577147.

Wan, Y., Luo, J., Zhang, Q., Wu, W., Huang, Y., and Zhou, M. (2018). Controller design for avoiding collisions in automated guided vehicle systems via labeled petri nets. *IFAC-PapersOnLine*, 51(7), 139 – 144. doi: https://doi.org/10.1016/j.ifacol.2018.06.292. 14th IFAC Workshop on Discrete Event Systems WODES 2018.

Appendix A. ELEMENTS OF DES THEORY

The dynamics of a DES is characterised by a set X of *states* or *modes*, i.e. all possible situations the system can be, a set E of events that trigger transitions from one state to another and finally, a transition function $f : X \times E \to X$ that describes the way these transitions are carried out. If $x_0 \in X$ is the initial state of the system and X_m is the set of *marked* states, that is the states that have some special significance for the process (e.g. final states), then the DFA describing the system is a five-tupple

$$G \triangleq (X, E, f, x_0, X_m)\ . \tag{A.1}$$

Definition 1. (Event string). A sequence of events $s = e_1e_2e_3\ldots$ of arbitrarily large length is called an event string.

Events and event strings can be concatenated. The zero element of this operation ε is called the *empty event*, for which it holds $\varepsilon\varepsilon\varepsilon\ldots\varepsilon s\varepsilon\varepsilon\varepsilon\ldots\varepsilon = s$.

Definition 2. (Language). A *language* L defined over an event set E is a set of finite-length strings formed from events in E.

Given two languages $L_a, L_b \subseteq E^*$, their concatenation is defined as

$$L_aL_b \triangleq \{s \in E^* \ :\ s = s_as_b,\ s_a \in L_a,\ s_b \in L_b\}\ .$$

Definition 3. (Kleene-closure of a set). The *Kleen-closure* E^* of an event set E is the countably infinite set of all finite (but arbitrarily long) strings of elements of E, including the empty string ε.

Therefore, the Kleene-closure L^* of a language $L \subseteq E^*$ over an event set E is defined as $L^* \triangleq \{\varepsilon\} \cup L \cup LL \cup \ldots$. If $s = tuv \in E^*$, then t, u, v are called *prefix*, *substring* and *suffix* of s, respectively.

Definition 4. (Prefix-closure of language). The *prefix closure* $\bar{L}$ of a language $L \subseteq E^*$ over an event set E is defined as

$$\bar{L} \triangleq \{s \in E^* \ :\ \exists t \in E^* \text{ such that } st \in L\ .\}$$

In general $L \subseteq \bar{L}$ and if $L = \bar{L}$, then L is called *prefix-closed.*

Definition 5. (Projection). The *projection* of a string s defined over the event set E_1 onto the event set E_2 is defined as

$$\begin{aligned} P_2(\varepsilon) &\triangleq \varepsilon \\ P_2(e) &\triangleq \begin{cases} e, & \\ \varepsilon & \text{if } e \in E_1 - E_2 \end{cases} \\ P_2(se) &\triangleq P_2(s)P_2(e) \text{ for } s \in E_1^*,\ e \in E_1 \end{aligned}$$

The projection of a language $L \subseteq E_1^*$ onto E_2 is defined as

$$P_2(L) \triangleq \{t \in E_2^* \ :\ \exists s \in L \mid P_2(s) = t\}\ .$$

Definition 6. Given an automaton $G = (X, E, f, x_0, X_m)$, a state $x \in X$ is said to be *coaccessible* to X_m if there is a path in the state transition diagram of G from state x to a marked state in X_m.

Appendix B. COLREGS - RULE 15

(Crossing situation). When two power-driven vessels are crossing so as to involve risk of collision, the vessel which has theother on her own starboard side shall keep out of the way and shall, if the circumstances of the case admit, avoid crossing ahead of the other vessel.