Anticipation of ship behaviours in multi-vessel scenarios

Dimitrios Papageorgiou[a,*], Peter Nicholas Hansen[a], Kjeld Dittmann[a] and Mogens Blanke[a]

[a]Automation and Control Group, Department of Electrical Engineering, Technical University of Denmark, Elektrovej 326, 2800 Kgs Lyngby, Denmark

ARTICLE INFO

Keywords:
Autonomous vessels
Finite-state automata
Behaviour anticipation
Situation awareness
COLREGS representation
Multi-vessel encounter
Dual-view assessment

ABSTRACT

Highly reliable situation awareness is a main driver to enhance safety via autonomous technology in the marine industry. Groundings, ship collisions and collisions with bridges illustrate the need for enhanced safety. Authority for a computer to suggest actions or to take command, would be able to avoid some accidents where human misjudgement was a core reason. Autonomous situation awareness need be conducted with extreme confidence to let a computer algorithm take command. The anticipation of how a situation develops is by far the most difficult step in situation awareness and anticipation is the subject of this article. The IMO International Regulations for Preventing Collisions at Sea (COLREGS), describe the regulatory behaviours of marine vessels relative to each other, and correct interpretation of situations is instrumental to safe navigation. Based on a breakdown of COLREGS rules, this article presents a framework to represent manoeuvring behaviours that are expected when all vessels obey the rules. The article uses a discrete-event systems framework of finite automata that segregate situation assessment from decision making. A framework is suggested that makes it possible to anticipate own ship and other vessels manoeuvring in a multi-vessel scenario. The framework is tested using realistic simulated scenarios.

1. Introduction

Safe navigation at sea requires proficient situation awareness, where navigators use a combination of perception, understanding and anticipation, based on careful outlook. Lack of situation awareness has been a root-cause of several collisions and groundings. Ongoing research is addressing the quest to develop computer algorithms that are reliable, predictable and can perform all elements of situation awareness. Awareness consists of perception, understanding and anticipation. Anticipation is the most challenging part, in particular in multi-vessel scenarios, and this is the subject of this research.

To obtain automated situation awareness, algorithms will process images from electro-optical sensors, radar and other instruments on board. They will interpret surroundings and predict behaviours of other traffic. When information confirms deviations from anticipated normal behaviours, algorithms shall predict imminent risks and provide decision support for safe manoeuvring. One aim is that results of risk assessment and risk avoidance can be presented as advice to a human navigator. Another is that this information can be presented to an autonomous manoeuvring system that can respond with a plan for immediate manoeuvring to avoid risk of collision or grounding.

Maritime safety is a constant source of concern, in particular the hazards that are related to human misinterpretation of intentions of surrounding traffic or erroneous interaction with the technology on board. Human errors on decisions cause most of the collisions at sea [1, 2]. With unattended navigation, where the navigator is on the bridge but has other tasks in parallel with navigation, or navigation with a navigator located at a Remote Control Center (RCC), reliable anticipation is a research topic of ever increasing importance [3].

Two main directions of research are being pursued to address the anticipation problem. The first one, motivated by the recent advances in behaviour inference of vehicles and pedestrians in the car industry sector [4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14], pertains to the use of machine learning based on statistics on observations of antecedent behaviors of vessels that carry an Automatic Identification System (AIS) transponder. Some of these learning-based methods employ clustering of past vessels trajectories to infer the current and future ones. Single Point Neighbour Search and Multiple Trajectory Extraction Methods was pursued in [15]. The authors followed a similar approach in [16], where they employed clustering based on Principal Components Analysis. A dual linear autoencoder scheme was used in [17] for predicting an entire trajectory of a ship. The method was based on utilising historical data for clustering and classifying past ship trajectories to infer future ones. The main idea related to historic trajectories of vessels that had been in the vicinity of the target vessel's current position. Another approach relates to predicting the future kinematics of the target vessel.Trajectory prediction based on estimation of ship dynamics was presented in [18]. The approach followed an exhaustive search and listing of all possible states of the vessels by assuming a fixed set of possible rudder angle values. Optimization was then employed to assess the risk of collision. Local clustering of AIS-inferred trajectories was used in [19] to create the anticipated path of a surface vessel, the latter being based on a background (also AIS-based) model. A Deep Neural Network (DNN) was used in [20] for estimating the trajectory of a Target Vessel (TV) based on AIS, Electronic Navigational Chart (ENC) and radar data. Least Squares regression-based curve fitting was used in [21] for

---

*Principal corresponding author. Tel.: +45 45253573

dimpa@elektro.dtu.dk (D. Papageorgiou); pnha@elektro.dtu.dk (P.N. Hansen); kjeditt@elektro.dtu.dk (K. Dittmann); mb@elektro.dtu.dk (M. Blanke)

ORCID(s): 0000-0002-4900-4083 (D. Papageorgiou); 0000-0001-9972-1649 (P.N. Hansen); 0000-0001-6263-6096 (K. Dittmann); 0000-0002-9181-9518 (M. Blanke)

generating a function of time that represented certain states of a system (e.g. position, velocity etc.). The resulting model was played forward in time to predict the future trajectories. An alternative research direction for anticipating vessels behaviours relates to analysis in real time of observed tracks of objects in an awareness zone around own vessel, using the COLREGS [22] and sea chart information. Automated COLREGS-based assessment of own vessel obligations has been reported in several publications. Anticipation in the form of track estimation using Integrated Probabilistic Data Association was employed in [23], where COLREGS were incorporated only for the generation of evasive manoeuvres. Prediction of future trajectories in short time horizon via integration of the dynamical models was pursued in [24] for target following. The COLREGS were considered only in the path planning phase. A COLREGS-compliant Discrete-Event Systems (DES) framework was proposed in [25, 26] for facilitating autonomous situation awareness in multiple vessel scenarios. Short-horizon intention inference of ships was explored in [27] for estimating collision avoidance policy of target vessels based on spacial ship information.

While the majority of literature treats anticipation of future situation in marine vessels as a problem of estimating the trajectories of the vessels in a short time horizon, this study also considers the *intention* of other TVs. Unlike the approach followed in [27], the intention inference is not limited to a short time horizon. The salient feature of this inclusion pertains to the element of COLREGS compliance, the latter encapsulating not only spacial and kinematic considerations but other parameters, such as the type of the vessel, the consistency of sensory instrumentation etc. This also reflects the way human navigators conclude about the behaviour of TVs. The proposed method significantly extends previous results [25], [26] by considering the interaction of Own Ship (OS) with TVs and that between different TVs. The article explains how this is obtained by employing a discrete-event systems (DES) framework that guarantees the prevention of deadlocks in the algorithm. Furthermore, a *dual-view assessment* principle is introduced that significantly reduces complexity of the multi-vessel anticipation task. Simulations of realistic multi-vessel encounters scenarios are carried out to illustrate the effectiveness of the proposed methods.

The paper is structured as follows: Section 2 introduces the architecture designed for autonomous situation awareness and its components. Section 3 details the DES-based framework for understanding and anticipation of TVs behaviours and the required actions of OS. Section 4 presents simulation results to assess the anticipation scheme and report performance of the method. Reflections on the results and extensions are provided in Section 5. Finally, conclusions are drawn in Section 6.

2. Anticipation for autonomous navigation

In a multi-vessel environment, anticipated reactions of other vessels is an essential piece of information to prevent accidents. If safe autonomous navigation is to be achieved, a framework that guarantees automated situation awareness needs to facilitate ship behaviour anticipation. Such a framework was introduced in [28], where the fundamental components of an autonomous vessel were detailed. The autonomous ship is operated by an internal intelligence abbreviated as the *autonomous supervisor*, which is responsible for watch-keeping, navigation and system monitoring. The autonomous supervisor consists of three main modules, namely, the Autonomous Coordination Supervisor (ACS), the Autonomous Navigation Supervisor (ANS) and the Autonomous Platform Supervisor (APS). The ACS is the main coordinating unit of the autonomous system, while the APS assesses the overall system health by gathering information from local diagnostic systems associated to each machinery component. The ANS is responsible for ensuring that all navigation is conducted in a safe manner. These modules are interconnected in a hierarchical distributed architecture that facilitates clear segregation of the different functionalities and services as shown in Figure 1. This is achieved via subscribe-publish software policy that ensures robust and modular communication of the different modules [29].

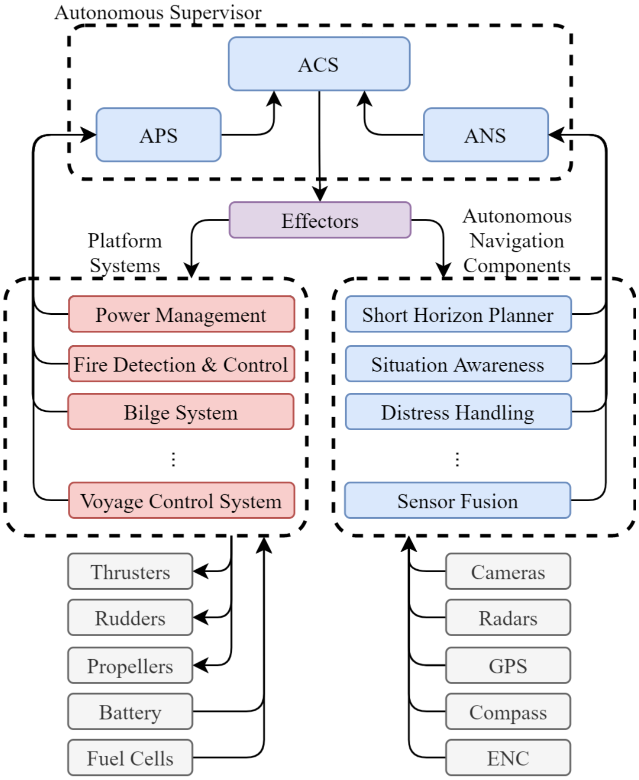


Figure 1: Ship autonomy architecture and allocation of functions.

The ANS has three main tasks:

1. Assess and interpret the current situation.
2. Anticipate the future situation within a time horizon.
3. Detect anomalies and deviant behaviours.

A dedicated module called the Situation Awareness Service (SAS), which directly reports to the ANS, is responsible for the first two tasks, which constitute the topic of this study. Anomaly detection will be treated in a separate study and is, thus, left outside the scope of this paper.

2.1. SAS architecture

As mentioned earlier, the SAS is tasked with describing the current and future situation based on the interpretation of the collected data from the cameras, radars, etc. that pertain to the surrounding environment of the vessel (encountered vessels, buoys, land structures etc.). This data comes from a sensor fusion module that publishes all the relevant information in a Consolidated Object List (COL) topic. Each detected object (vessel or buoy) that lies within a region of predetermined size called the *awareness zone* of the OS can be of navigational interest and is therefore added to an *awareness list*. A sequence of tests is then performed for each object, which corresponds to the action chain of an experienced human navigator for understanding the current situation. The outcome is interpreted based on the COLREGS and a high-level report is given along with a recommendation concerning whether the OS should give way or stand on. This procedure was detailed in [25] and expanded in [26].

If the foregoing process is repeated but now from the perspective of the TVs (viewing each TV as the OS), then the former recommendations correspond to the *intended actions* of the TVs. This, combined with the trajectories (in the form of a list of waypoints) for each surrounding vessel constitutes the *estimated behaviour* of each TV within a given time horizon. This description is formalised in the following definition.

Definition 1 (Estimated behaviour). Consider a time interval (horizon) $\mathcal{H} = [t_0, t_1]$ and a set of n moving agents (vessels) and m stationary agents (buoys, debris, obstacles, unknown objects etc.). The estimated behaviour of a vessel j within $\mathcal{H}$ is defined as the pair $\mathcal{B}_j \triangleq (I_j(t_1), W_j)$, where $I_j(t)$ is a vector of $m + n$ elements, each of which is a boolean-valued function of time $i_{jk}(t) \in \{0, 1\}$ denoting whether the vessel j will "stand on" or "give way", respectively with respect to the object k at $t \in \mathcal{H}$. The list $W_j = \{w_1, w_2, \ldots, w_N\}$ contains N waypoints that describe the estimated position of the vessel at N time instances up to time $t = t_1$.

From the foregoing definition, it is clear that if t_1 corresponds to the current time, the estimated behaviour of all vessels in the awareness list constitutes a description of the current situation at sea. On the other hand, if t_0 represents the current time and $t_1 = t_0 + \Delta t$, where Δt is a positive time offset, the corresponding estimated behaviours give an overview of the *anticipated situation*.

Figure 1 illustrates the architecture of the ship's autonomy system that relates to situation awareness. Its central module is the SAS, which consists of two internal sub-modules. The first one is called "Intention Estimation" and outputs the intended actions of all the vessels within a list (COL or a list provided by the ANS) together with an encoded report that contains the details of the associated navigation tests. If these tests concern the current time, the SAS publishes at the "Situation" data topic, to which both the ACS and the ANS subscribe. Otherwise, the output from the SAS is published to the "Anticipated Situation" topic. The second subsystem of SAS is the "Trajectory Estimation", which for current time provides the position of the vessel taken by the COL, whereas for a future time instant it estimates the vessel trajectory up to that instant in the form of a number of waypoints. This sub-module publishes to the "Anticipated Trajectories" topic. Both SAS sub-modules are detailed in Section 3. The SAS subscribes to the following topics:

- OS data: OS information (position , speed etc.).
- COL: information on surrounding vessels and objects (relative position, speed, type, etc.).
- Anticipated COL: projected information on surrounding vessels and objects (relative position, speed, type, etc.) in the future.
- Understanding config: Configuration data for performing the navigational tests (Closest Point of Approach (CPA), Time for Closest Point of Approach (TCPA) limits etc.).
- Anticipation config: Configuration data for performing the navigational tests in the future (vessel types, look-out radius, maximum anticipation time, CPA, TCPA limits etc.).
- Anticipated situation request: Specified anticipation request properties (time horizon, time of request, etc.).

3. Ship behaviour estimation

This section details the structure and elements of the two SAS sub-modules for estimating the current or future behaviour of the vessels in a given navigation scenario. The proposed scheme utilises a modular architecture comprising a DES-based framework for estimating the intended action of the vessels and a trajectory estimator based on the kinematic models of the vessels. The modularity of the two sub-modules allows for independent extension of the methods.

3.1. Intention estimation

Estimation of the vessels' intended actions is achieved via a DES-based framework, which comprises two different types of Deterministic Finite-State Automata (DFA) [30]. The first type includes several instances of the *Understanding automaton*. Each instance facilitates the sequential tests that are performed by a navigator in order to assess the situation at sea with respect to a specific object (e.g. whether the object is a vessel or a buoy, the vessel is motor-driven or sailboat, the CPA etc.). These tests correspond to actual navigation loops as described in [25, 26] and are separately carried out for each object.

The *Coordinator automaton* is responsible for handling information coming from the COL (e.g. regarding the detection of a new object) and for managing reports obtained

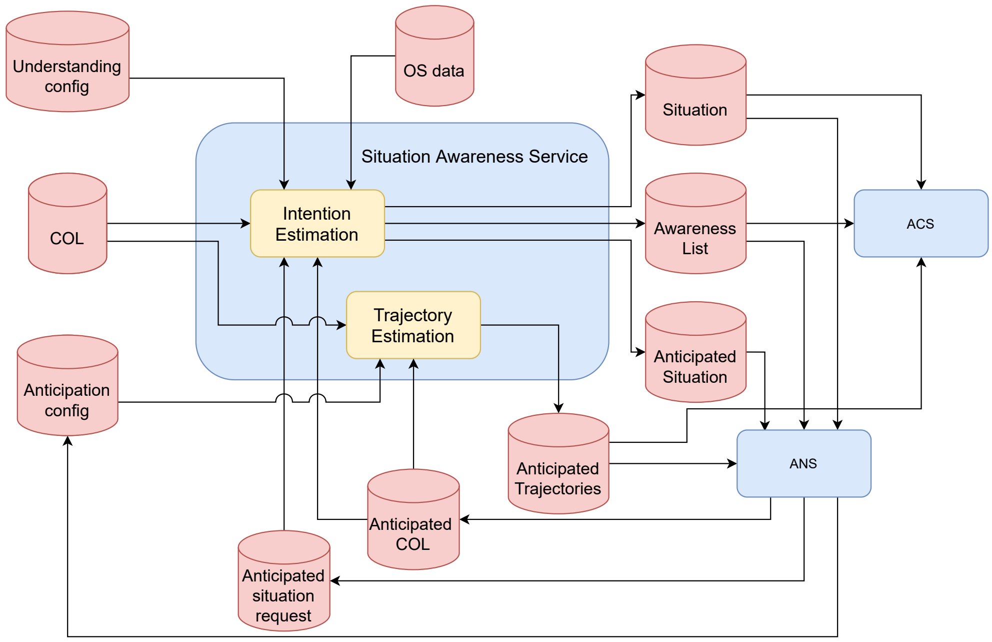


Figure 2: SAS - Module architecture and interfaces. The cylinders denote different data topics to which SAS publishes or subscribes.

from the different understanding automata. Each time a new object is detected, the coordinator automaton generates an instance of the understanding automaton for assessing the situation with respect to the object from the OS perspective. Upon completion of the sequential tests, the understanding automaton reports back to the coordinator automaton the test results. These, are then interpreted by the coordinator automaton and associated to the value of the function $i_{jk}(t) \in \{0, 1\}$, which describes the intention of the OS to either stand on or give way to the examined object. This interpretation is made via the *COLREGS service*, which is a look-up table implementing Rules 7 and 13-17 of the COLREGS (see Appendix B). Since the intention of the TVs is also needed, the foregoing procedure is carried out for the TVs as well. The final output of the coordinator automaton is the Situation Array (SA), which has n rows and $m+n$ columns with n being the number of moving agents, i.e. OS and TVs and m being the number of stationary agents (buoys, debris, obstacles, unknown objects etc.). The first n columns of the SA correspond to the OS and the TVs, while the next m columns to the stationary agents. The rows only correspond to the moving agents. The (j, k)-element of the array, denoted here by $a_{jk}(t)$, represents the intended action of the j^{th} vessel with respect to the k^{th} object (moving or stationary) at time t and it is defined as

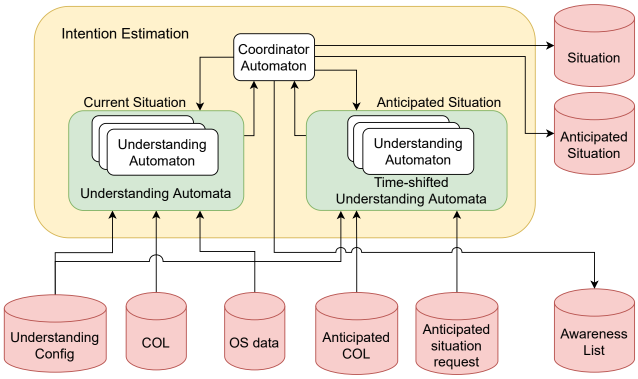


Figure 3: SAS - Intention anticipation. The cylinders denote different data topics to which SAS publishes or subscribes.

$$a_{jk}(t) = \begin{cases} 0 & \text{if } j = k \\ i_{jk}(t) & \text{if } j \neq k \end{cases}, \ (j,k) \in \{1 \dots n\} \times \{1 \dots (m+n)\} \tag{1}$$

since a vessel may only attempt an evasive manoeuvre in case of an encounter with another object. Table 1 illustrates an example of a SA in a scenario with n TVs ($n+1$ moving agents) and m buoys (stationary agents). The subsequent subsections provide a detailed description of the coordinator and understanding automata and introduce the idea of *dual-view assessment* for reducing complexity.

Remark 1. In the following, the term *report agent* will refer to the vessel that performs an assessment of a situation with respect to an object, which can be either another vessel (moving agent) or a stationary object (buoys, debris, etc.).

3.1.1. Coordinator automaton

The coordinator automaton is defined as the five-tuple

$$G_c \triangleq (C, E_c, f_c, C_1, C_1) \ . \tag{2}$$

where $C \triangleq \bigcup_{i=1}^{14} \{C_i\}$ is the set of possible states listed below:

Table 1
Example of a Situation Array for a case of n vessels and m objects. The diagonal elements of the left $n \times n$ block are always 0.

-	OS	TV_1	TV_2	$\cdots$	TV_{n-1}	O_1	$\cdots$	O_m
OS	0	0	1	$\cdots$	1	1	$\cdots$	0
TV_1	1	0	1	$\cdots$	0	0	$\cdots$	0
TV_2	1	0	0	$\cdots$	1	0	$\cdots$	1
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\ddots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
TV_{n-1}	0	1	0	$\cdots$	0	1	$\cdots$	0

$n\times n$ block $n\times m$ block

C_1: Wait for new reports (initial and marked state).

C_2: Check type of new report.

C_3: Check if detected object is already being tracked.

C_4: Check if object type has changed.

C_5: Update object type.

C_6: Check if report agent is OS.

C_7: Generate new instance of understanding automaton associated with the unique ID of the tracked object (from OS to object), and append the object to the awareness list.

C_8: Calculate and compare associated risk to highest recorded risk. Update awareness list.

C_9: Check if reported object is a vessel (new object).

C_{10}: Generate report based on dual-view assessment.

C_{11}: Call COLREGS service, update and output Situation Array.

C_{12}: Check if reported object is a vessel (existing object).

C_{13}: Calculate relative configuration of the objects and generate the rest of the automata (report agent TV with respect to all other objects, except for OS).

C_{14}: Check whether reporting vessel is power-driven.

The event set $E_c \triangleq \bigcup_{i=0}^{13}\{c_i, \neg c_i\}$ is the set of events that may trigger transitions between two states of the coordinator, where e_i, $i = 1, \dots, 13$ are given in the following list:

c_1 : A new report is available.

c_2 : Report was sent by understanding automaton.

c_3 : Detected object is currently being tracked.

c_4 : Different object type from that on awareness list (object type has changed).

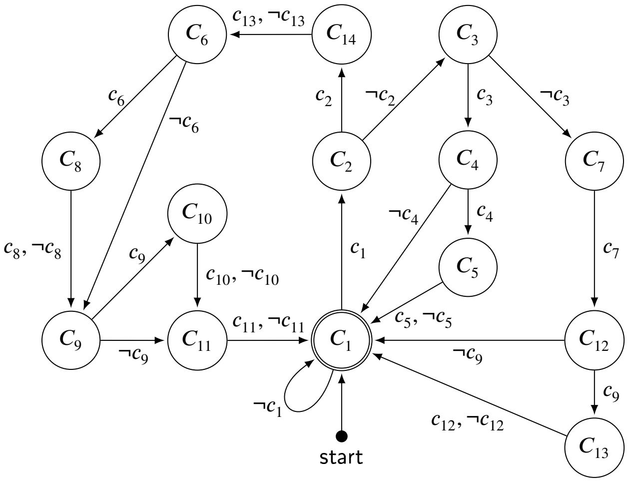


Figure 4: State transition diagram of the coordinator automaton.

c_5 : Objected type has been updated on awareness list.

c_6 : Report agent is OS.

c_7 : Relevant understanding automaton instance (OS to detected object) has been generated, object appended to awareness list.

c_8 : Collision risk has been calculated and the awareness list has been updated.

c_9 : The object is a vessel.

c_{10} : Report based on dual-view assessment has been generated.

c_{11} : The Situation Array has been updated and sent.

c_{12} : Remaining automata have been generated.

c_{13} : Reporting vessel is power-driven.

The state transition diagram of the coordinator automaton is shown in Figure 4 and it can completely define the extended state transition function $f_c : \mathcal{C} \times E_c^* \to \mathcal{C}$. The languages generated and marked by G_c are defined as

$$\mathcal{L}(G_c) \triangleq \left\{ s \in E_c^* \;:\; f_c(C_1, s) \text{ is defined} \right\}$$
$$\mathcal{L}_m(G_c) \triangleq \left\{ s \in E_c^* \;:\; f_c(C_1, s) = C_1 \right\}$$

where s is a string of events in the event set of the automaton.

Remark 2. The coordinator automaton is non-blocking since all its states are coaccessible, i.e. there is a path from every node in the transition diagram of G_c to the marked state [30].

Remark 3. Since the positions and relative bearings of each detected object with respect to the OS are known, calculating the relative position and bearing of the objects with respect to every TV amounts to applying a coordinates transformation of a transition and a rotation.

Remark 4. The risk calculation in C_8 can be done based on the CPA, TCPA and the Distance from the Closest Point of Approach (DCPA) as suggested in [31].

3.1.2. Understanding automaton

As mentioned earlier, several instances of the understanding automaton can be generated by the coordinator, each of which corresponds to assessing the situation from the perspective of a vessel (OS or TV) with respect to a detected object (moving or stationary agent). The set $\mathcal{U} \triangleq \bigcup_{i=1}^{13}\{U_i\}$ of the discrete states of an instance of the understanding automaton is enumerated below [26] (additional nomenclature is provided in Table 2):

U_1: Compare CPA to d_{req} (initial and marked state).

U_2: Compare TCPA to t_{req}.

U_3: Evaluate object type.

U_4: Calculate own ship bearing relatively to target vessel.

U_5: Assess target vessel type.

U_6: Compare TCPA to Time to Next Waypoint (TTW).

U_7: Check if next waypoint is towards target vessel.

U_8: Compare CPA to distance action limits d_{act}.

U_9: Compare TCPA to time action limits t_{act}.

U_{10}: Validate geometry of detected object using Electronic Chart Display and Information System (ECDIS).

U_{11}: Evaluate own ship position uncertainty.

U_{12}: Compare next waypoint to detected object's position.

U_{13}: Report events sequence to coordinator (marked state).

The event set associated to the discrete states set $\mathcal{U}$ is defined as $E_u \triangleq \bigcup_{i=0}^{17}\{u_i, \neg u_i\}$ where

u_1 : Object violates CPA awareness level (CPA$< d_{req}$).

u_2 : Object violates TCPA awareness level (TCPA$< t_{req}$).

u_3 : Detected object is a vessel.

u_4 : Detected object overtaking.

u_5 : Detected object overtaken.

u_6 : Detected object is head-on towards own ship.

u_7 : Detected object is on port side.

u_8 : Detected object is on starboard side.

u_9 : Detected object is power-driven.

u_{10} : TCPA larger than TTW.

Table 2
Explanatory nomenclature for G_u.

Symbol	Explanation
d_{req}	distance limit for assessing detected objects
t_{req}	time limit for assessing objects with CPA$\leq d_{req}$
d_{act}	distance limit for taking (manoeuvring) action
t_{act}	time limit for taking (manoeuvring) action

u_{11} : Object geometry matches ECDIS description.

u_{12} : Own ship position certainty factor above threshold.

u_{13} : Conflict between route and land/buoy position.

u_{14} : Next waypoint on the same side as object.

u_{15} : CPA smaller than distance action limits d_{act}.

u_{16} : TCPA smaller than time action limits t_{act}.

u_{17} : Report sent to coordinator.

The extended transition function $f_u : \mathcal{U} \times E_u^* \rightarrow \mathcal{U}$ is fully described by the state transition diagram in Figure 5. Based on the foregoing description, the understanding automaton G_d is defined as the five-tuple

$$G_u \triangleq (\mathcal{U}, E_u, f_u, U_1, \{U_1, U_{13}\}) . \tag{3}$$

The set of marked states for G_u is chosen to contain only the states U_1, U_{13} because they signify the end of a full assessment cycle and the reporting to the coordinator, respectively. The languages generated and marked by G_u are defined as

$$\begin{aligned} \mathcal{L}(G_u) &\triangleq \left\{ s \in E_u^* \ : \ f_u(U_1, s) \text{ is defined} \right\} \\ \mathcal{L}_m(G_u) &\triangleq \left\{ s \in \mathcal{L}(G_u) \ : \ f_u(U_1, s) \in \{U_1, U_{13}\} \right\} . \end{aligned}$$

Remark 5. The automaton G_u is also non-blocking since all its states are coaccessible. Moreover, the combination of the coordinator automaton with the generated instances of the understanding automaton constitutes the parallel composition of a finite number of non-blocking automata and it is, therefore, itself non-blocking [30].

Remark 6. A transition from U_4 to U_5 is possible to be triggered only if one of the mutually exclusive events u_4, u_5, u_6, u_7, u_8 occurs. This implies that one, and only one, of the events will occur, e.g. $u_4 \Rightarrow \neg u_5 \wedge \neg u_6 \wedge \neg u_7 \wedge \neg u_8$.

Remark 7. Although the transition from state U_5 to U_6 does not depend on whether the target vessel is power driven or not, different COLREGS rules apply for non-power driven vessels, hence the need for the distinction between the two types.

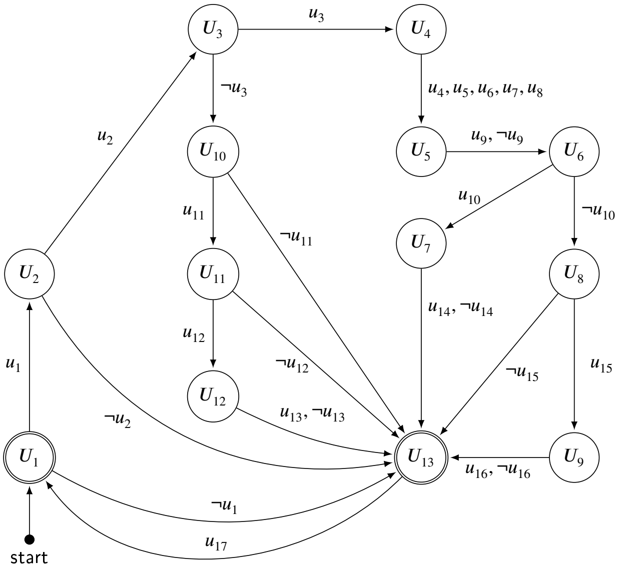


Figure 5: State transition diagram of the understanding automaton.

3.1.3. COLREGS interpreter

The COLREGS interpreter comprises an agent-agnostic function $f_{CLR} : \mathcal{L}_m(G_u)\mathcal{L}_m(G_c) \rightarrow \{0,1\}$ that associates a sequence of events (a word) to the set $\{0,1\}$, where "1" denotes that the agent intends to give way and "0" denotes that it will stand on. Define the sets

$$A \triangleq E_u - \bigcup_i \{u_i, \neg u_i\},\ i \in \{10, 14, 15, 16, 17\} \tag{4}$$

$$B \triangleq \{c_{13}, \neg c_{13}\} \subset E_c \tag{5}$$

and a word $s \in \mathcal{L}_m(G_u)\mathcal{L}_m(G_c)$. Furthermore, define the set of words R according to the following enumeration:

$$\begin{aligned} R \triangleq \{ & u_1u_2u_3u_5u_9c_{13},\ u_1u_2u_3u_5\neg u_9c_{13}, \\ & u_1u_2u_3u_5u_9\neg c_{13},\ u_1u_2u_3u_5\neg u_9\neg c_{13}, \\ & u_1u_2u_3u_6u_9c_{13},\ u_1u_2u_3u_6\neg u_9c_{13}, \\ & u_1u_2u_3u_6u_9\neg c_{13},\ u_1u_2u_3u_6\neg u_9\neg c_{13}, \\ & u_1u_2u_3u_8u_9c_{13},\ u_1u_2u_3u_8\neg u_9c_{13}, \\ & u_1u_2u_3u_7\neg u_9c_{13},\ u_1u_2u_3u_8\neg u_9\neg c_{13}, \\ & u_1u_2\neg u_3u_{11}u_{13}c_{13},\ u_1u_2\neg u_3u_{11}u_{13}\neg c_{13} \} \end{aligned}$$

The function f_{CLR} is then defined as

$$f_{CLR}(s) = \begin{cases} 1, & P_A(s)P_B(s) \in R \\ 0, & \text{otherwise} \end{cases} \tag{6}$$

where $P_A(s)P_B(s)$ is the concatenation of the *projections* (see Appendix A) of the word s onto the sets A and B, respectively. The set R is an encoding of all the possible COLREGS-described encounters between two agents (vessel-to-vessel or vessel-to-object) into strings of events (words) belonging to the event sets of the coordinator and understanding automata. What the projections P_A, P_B essentially do, is removing from the reported words all the events that are not needed for inferring the intention of a vessel based on the COLREGS.

The COLREGS interpreter can be implemented as a look-up table that associates each word coming from the understanding and coordinator automata to a COLREGS case and can be updated off-line. According to this table, each word can be decoded to a high-level text description of the encountered situation as shown below for the word $s = u_1u_2u_3u_8u_9 \neg u_{10}u_{15}u_{16}u_{17}c_{13}c_6c_8c_9$:

$$P_A(s)P_B(s) = u_1u_2u_3\overbrace{u_8}^{\text{Object on starboard side}}\underbrace{u_9}_{\text{Target is power-driven}}\overbrace{c_{13}}^{\text{Own ship is power-driven}} \rightarrow \text{Starboard crossing of power-driven vessel}$$

This decoding of the COLREGS encounters can be useful when the current or anticipated situation is communicated to the human proxy (e.g. the remote control center) via the ACS.

3.1.4. Complexity and dual-view assessment

As discussed earlier, obtaining a full overview of the current or anticipated situation at sea requires the combined assessment of all vessels with respect to all detected objects. Fulfilling this requirement in a multiple-vessels scenario does not scale well in terms of complexity. Indeed, in a case of n vessels, including the OS, and m stationary agents, the total number of understanding automata that need be generated by the coordinator automaton is $N = n(n + m - 1)$. This is because each vessel has to assess the situation with respect to every detected object except for itself. However, a significant amount of information regarding two agents can actually be extracted by only examining the assessment of one of them. The following example illustrates this idea of dual-view assessment.

Consider the case of a power-driven OS encountering an also power-driven TV that attempts to cross from starboard. Assume that the CPA and TCPA limits are violated and there is a risk for collision. Then according to the COLREGS, the OS must perform an evasive manoeuvre, i.e. it needs to give way. Assume also that the word reported by the understanding automaton that corresponds to the OS assessment with respect to the TV is $s_1 = u_1u_2u_3u_8u_9\neg u_{10}u_{15}u_{16}u_{17}c_{13}c_6c_8c_9$ (also appending the coordinator word up to state C_{10}). Then its projection onto the sets P_A, P_B is given by

$$r_1 \triangleq P_A(s_1)P_B(s_1) = u_1u_2u_3u_8u_9c_{13}$$

By examining now the TV's assessment s_2 of the situation with respect to the OS, it is easy to observe that its projection onto P_A, P_B is given by

$$r_2 \triangleq P_A(s_2)P_B(s_2) = u_1u_2u_3u_7u_9c_{13}$$

The main idea of the dual-view assessment is that r_2 can be directly constructed from r_1, *without the need of generating one additional automaton*. Specifically, the first three events in r_2 need to be the same with those in r_1 since they describe

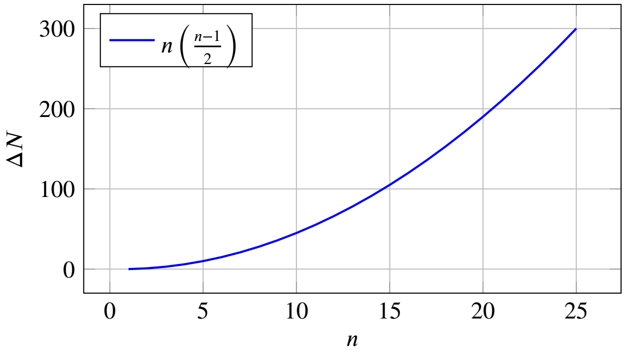


Figure 6: Number of understanding automata that need not be generated under dual-view assessment, given as a function of the number of vessels.

a bilateral agent relation (i.e. the CPA and TCPA limits are violated for both vessels and it is a vessel-to-vessel consideration). Moreover, if the TV is on the starboard of the OS, then the latter is on the port side of the TV, therefore u_8 needs to be replaced with u_7 in r_2. In general, the third event of r_2 can be obtained by the third event in r_1 according to the rule:

$$u_4 \leftrightarrow u_5 \text{ (Overtaking} \leftrightarrow \text{Overtaken)}$$
$$u_6 \leftrightarrow u_6 \text{ (Head-on} \leftrightarrow \text{Head-on)}$$
$$u_7 \leftrightarrow u_8 \text{ (Port} \leftrightarrow \text{Starboard)}$$

Finally, the last two events in r_2 concern the type of the reporting and assessed vessel, respectively and they can be obtained from r_1 by applying the following rule:

$$u_9 \in r_1 \Rightarrow c_{13} \in r_2$$
$$\neg u_9 \in r_1 \Rightarrow \neg c_{13} \in r_2$$
$$c_{13} \in r_1 \Rightarrow u_9 \in r_2$$
$$\neg c_{13} \in r_1 \Rightarrow \neg u_9 \in r_2$$

It follows that under the dual-view assessment the total number of required understanding automata is given by

$$N = nm + \binom{n}{2} = nm + \frac{n!}{2!(n-2)!} = n\left(m + \frac{n-1}{2}\right) \quad (7)$$

This implies that applying the dual-view assessment principle allows for a reduction of the required number of automata that quadratically increases with the number of vessels, i.e.

$$\Delta N = n\frac{n-1}{2} = O(n^2)\ ,$$

which, as Figure 6 shows, can be quite significant for a large number of vessels . Figure 7 illustrates the complexity difference in terms of number of required automata as a function of the total number of vessels and stationary agents in a situation. An overview of the procedure of reconstructing r_2 from s_1 is given in Algorithm 1.

3.2. Trajectory estimation

The main advantage of the considered architecture for the SAS pertains to the separability of the interpretation of the situation at sea from the estimation of the vessels' trajectories. The latter can either be learning-based as pursued for example in [15, 17], where AIS data is used or it can be model-based (e.g. by integrating current measurements of position and velocity). The default output of the trajectory estimation sub-module is the latest position of the vessel directly taken by the COL.

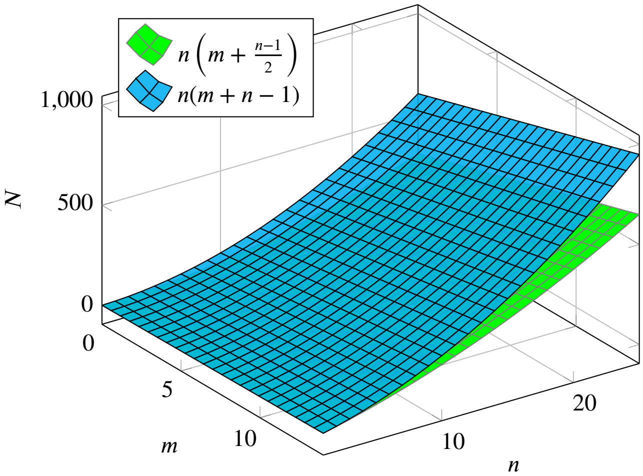


Figure 7: Number of generated understanding automata with (green) and without (blue) the dual-view assessment principle.

When a trajectory anticipation request is made by the ANS within a given time horizon, the sub-module outputs a list of waypoints based on a selected segmentation of the given time interval. The last waypoint corresponds on the projected position at the last time instant of the horizon. The method of calculating the projected trajectories of the vessels in this study is based on integration of the kinematics assuming constant speed and course. The simplicity of the method allows for fast estimation but at the cost of short time horizons due to the validity limitations of the constant speed and course assumptions. Figure 8 illustrates the trajectory estimation scheme considered in this paper for the case of future trajectories anticipation. The segmentation of the estimated trajectory can be done based on a predefined selection of number of waypoints. For the selected method of trajectory prediction, the larger this number is, the more robust is the trajectory estimation against changes in the speed and course of the target vessel. As already mentioned, this also depends on the size of the prediction horizon, which can be selected as the smallest (positive) TCPA between OS and each TV in the awareness zone. Such choice can be related to risk assessment and prioritisation that is often correlated to the TCPA.

4. Simulations

4.1. Simulation scenarios

The situation awareness and anticipation framework was tested in simulation in three different scenarios. All cases concerned open-waters navigation with vessels being the only

Algorithm 1 Dual-view assessment between TV$_1$ and TV$_2$

```
Input: $s_1$ % Word from TV$_1$ understanding and coordinator
    automaton
Output: $r_2$ % Projected word for TV$_2$
  $r_2 \leftarrow \varepsilon$ % Initialise $r_2$
  Define $C = \{u_1, \neg u_1, u_2, \neg u_2, u_3, \neg u_3\}$
  Define $D = \{u_4, u_5, u_6, u_7, u_8\}$
  $r_2 \leftarrow r_2 P_C(P_A(s_1) P_B(s_1))$ % $A$, $B$ are defined in (4), (5)
  switch $P_D(r_1)$ do
      case $u_4$
          $r_2 \leftarrow r_2 u_5$ % (Overtaking ← Overtaken)
      case $u_5$
          $r_2 \leftarrow r_2 u_4$ % (Overtaken ← Overtaking)
      case $u_6$
          $r_2 \leftarrow r_2 u_6$ % (Head-on ← Head-on)
      case $u_7$
          $r_2 \leftarrow r_2 u_8$ % (Port ← Starboard)
      case $u_8$
          $r_2 \leftarrow r_2 u_7$ % (Starboard ← Port)
  if $c_{13} \in r_1$ then
      $r_2 \leftarrow r_2 u_9$
  else
      if $\neg c_{13} \in r_1$ then
          $r_2 \leftarrow r_2 \neg u_9$
      end if
  end if
  if $u_9 \in r_1$ then
      $r_2 \leftarrow r_2 c_{13}$
  else
      if $\neg u_9 \in r_1$ then
          $r_2 \leftarrow r_2 c_{13}$
      end if
  end if
  return $r_2$
```

agents. All ships were assumed to be power-driven with the same size of awareness zone. The first two scenarios considered the task of anticipating the situation 20 minutes in the future, while the third one emphasised on the situation assessment in a six-vessel encounter, especially in relation with the complexity of the task. In each scenario, the position of the OS is considered as the origin of the North-East coordinate system, where all distances are expressed in nautical miles (Nmi). Moreover, speed is measured in knots (kts), heading in the clockwise direction from North in degrees (deg) and time in minutes (min). The scenarios are detailed below.

4.1.1. Scenario 1

In the first scenario the OS detects two TVs in a configuration as shown in Figure 9(a). The positions and velocities of the three vessels are given in Table 3. All three vessels keep constant course and speed.

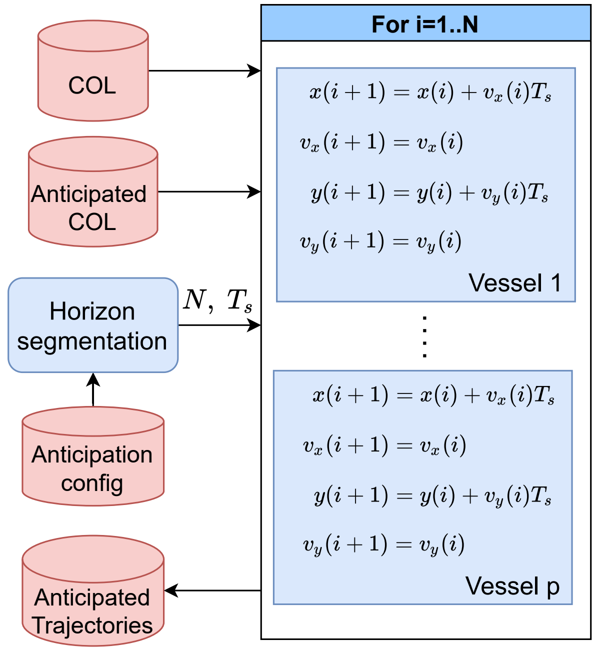


Figure 8: Trajectory estimation sub-module. The chosen method is conventional integration of the vessels' kinematics assuming the constant speed and course model.

Table 3
Scenarios 1 and 2 spatial configuration and kinematics.

Vessel	(North, East)	Speed	Course
OS	(0, 0)	11.5	0
TV$_1$	(4, 6.5)	18.5	270
TV$_2$	(8, 0)	11.5	180

4.1.2. Scenario 2

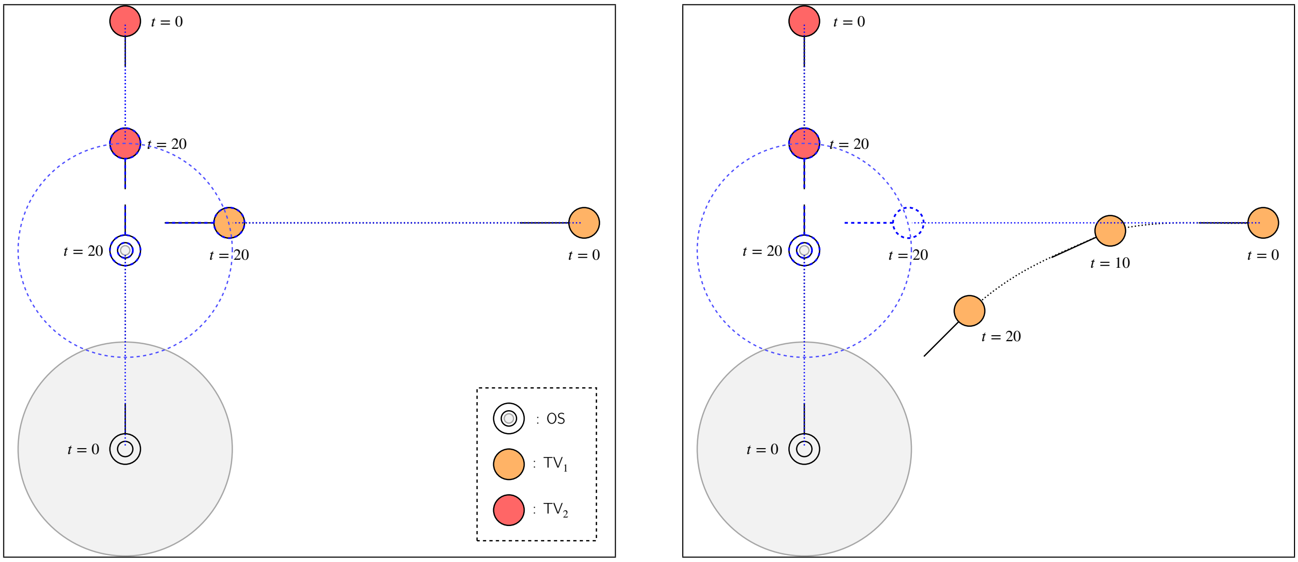
The second scenario is very similar to the first one. The same vessels and configuration are considered but now, the assumption of constant course and speed is violated for the case of TV$_1$. Specifically, after 10 minutes, the heading of TV$_1$ becomes 250 deg as illustrated in Figure 9(b). This will impact the validity of the anticipated behaviour for this vessel.

4.1.3. Scenario 3

In this scenario six targets are placed in a configuration as shown in Figure 10. The purpose of this test is to highlight how the proposed framework handles the complexity of obtaining an overall assessment of the situation, given all the tracked vessels. Although, all detected TVs are withing the awareness zone of the OS, not all of them pose a risk of collision. Table 4 details the configuration of the vessels in space as well as their velocities.

4.2. Results

In scenarios 1 and 2, the initial assessment of the situation can be characterised by a 3×3 awareness array with all its elements being 0. This is due to the fact that the vessels are very far from each other with no collision risk present.



(a) Scenario 1: Two-vessel encounter situation with constant course and speed.

(b) Scenario 2: Two-vessel encounter situation with constant speed and change of TV1 course.

Figure 9: Simulation scenarios 1 (left) and 2 (right) : Solid lines denote actual trajectories and positions, while blue dashed lines correspond to anticipated ones. Anticipation inconsistency occurs when the assumption for constant speed and course does not hold.

Table 4
Scenario 3 (multi-vessel) spatial configuration and kinematics.

Vessel	(North, East)	Speed	Course
OS	(0, 0)	12.5	0
TV$_1$	(5, 4)	12.5	225
TV$_2$	(-3, 3)	18.5	335
TV$_3$	(-4, -4)	12.5	315
TV$_4$	(-3, -8)	18.5	90
TV$_5$	(3, -4)	18.5	180
TV$_6$	(7, -3)	18.5	135

Table 5 shows the anticipated situation at time $t = 20$ min as this was projected at time $t = 0$ and the evaluation of the actual situation at $t = 20$ min. As expected, these two assessments match since the assumption of constant course and speed was true for the entire test. The colour convention in the table denotes intention calculated by generated understanding automata (orange) and obtained via dual-view assessment (green). Following the notation in Definition 1 and a segmentation of the time horizon in 5 instances, the behaviour of each of the TVs is given by

$$\mathcal{B}_1 = \left(\begin{bmatrix}0 & 0 & 1\end{bmatrix}, W_1\right)$$

$$W_1 = \left\{ \begin{bmatrix}4\\6.5\end{bmatrix}, \begin{bmatrix}4\\5\end{bmatrix}, \begin{bmatrix}4\\3.4\end{bmatrix}, \begin{bmatrix}4\\1.9\end{bmatrix}, \begin{bmatrix}4\\0.3\end{bmatrix} \right\}$$

$$\mathcal{B}_2 = \left(\begin{bmatrix}1 & 0 & 0\end{bmatrix}, W_2\right)$$

$$W_2 = \left\{ \begin{bmatrix}8\\0\end{bmatrix}, \begin{bmatrix}7\\0\end{bmatrix}, \begin{bmatrix}6.1\\0\end{bmatrix}, \begin{bmatrix}5.1\\0\end{bmatrix}, \begin{bmatrix}4.1\\0\end{bmatrix} \right\}$$

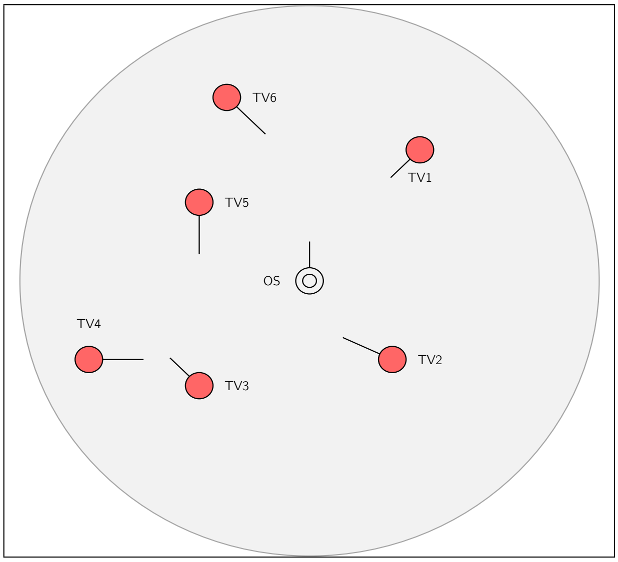


Figure 10: Simulation scenario 3 (multi-vessel encounter): Although all vessels are within the awareness zone of OS, not all of them pose a risk for collision.

The waypoints in W_1, W_2 were calculated by evaluating the kinematic models of the TVs at the 5 time instances within the time horizon.

Table 6 shows the outcome of the anticipation algorithm for the second scenario. Here there is a discrepancy between the anticipated situation obtained at $t = 0$ and the assessment of the actual situation at $t = 20$ min, which is indicated with the red colour in the anticipation array. The inconsistency occurs only for TV$_1$, which is not surprising since it is the

Table 5
Simulation scenario 1: situation/anticipation arrays for $t = 20$.

	Situation			Anticipation		
-	OS	TV$_1$	TV$_2$	OS	TV$_1$	TV$_2$
OS	0	1	1	0	1	1
TV$_1$	0	0	1	0	0	1
TV$_2$	1	0	0	1	0	0

Table 6
Simulation scenario 2 situation/anticipation arrays for $t = 20$.

	Situation			Anticipation		
-	OS	TV$_1$	TV$_2$	OS	TV$_1$	TV$_2$
OS	0	0	1	0	1	1
TV$_1$	0	0	1	0	0	1
TV$_2$	1	0	0	1	0	0

Table 7
Simulation scenario 3 situation array.

-	OS	TV$_1$	TV$_2$	TV$_3$	TV$_4$	TV$_5$	TV$_6$
OS	0	1	0	0	0	0	0
TV$_1$	0	0	0	0	0	0	1
TV$_2$	1	1	0	0	0	0	0
TV$_3$	0	0	0	0	0	0	0
TV$_4$	0	0	0	1	0	0	0
TV$_5$	0	0	0	0	1	0	0
TV$_6$	1	0	1	0	0	0	0

vessel that changes course at $t = 10$ min, violating the assumption on constant course. This also supports the idea that shorter time horizon anticipation can compensate to some degree for uncertainties in the tracked vessel kinematics.

Finally, the situation assessment for Scenario 3 is shown in Table 7. The total number of required automata for a complete assessment of the situation is $n(n-1) = 7 \cdot 6 = 42$. However, employing the dual-view assessment principle according to (7) reduced this number to $N = \frac{n(n-1)}{2} = 21$. This reduction obviously implies lower computational cost and faster situation assessment. The reports generated by the automata and the associated dual-view assessments are shown in Table 8 along with the output of the COLREGS interpreter function.

5. Discussion

The simulation tests presented in the previous section illustrated the applicability of the proposed framework for situation assessment and anticipation of ships future behaviours. It interesting to note that the method provides an overall (current of future) situation assessment, regardless of whether all vessels directly interact with the OS. Although the method is not exclusively based on trajectory estimation of the TVs, it is evident that the latter determines the quality of future projections of the situation at sea. In principle, the accuracy of the anticipated situation depends on the accuracy of the estimated future trajectories. However, this means that reduced capability in predicting future kinematics only limits the span of the time horizon within which the prediction is made. The example in scenario 2 showed that situation anticipation can be accurately carried out up to $t = 10$ min. It should be noted that this limitation is only due to the selection of the kinematics-based trajectory prediction. The modular architecture of the proposed situation awareness scheme allows for integrating a different method for trajectory estimation, e.g. one that employs learning-based clustering.

Additional inconsistencies between actual and anticipated situation may also arise from the uncertainty in the OS perception system. A characteristic example is the problem of deciding between head-on and port crossing situation in the case when the relative bearing between two vessels is close to the border of these two regimes. The perception uncertainty also affects the effectiveness of the dual-view assessment. Introducing stochastic representations of the event-driven dynamics of the method based on the uncertainty in the OS measurements can be one way of addressing this issue.

As a final remark, it should be mentioned that the proposed framework covers all the COLREGS rules detailed in Appendix A, with a notable exception of rule 9. To facilitate the applicability of rule 9, the system must be able to assess if a target is restricted in terms of manoeuvrability. Such an assessment requires two considerations, i.e. the overall manoeuvrability capabilities of a target, and the manoeuvrability restrictions imposed by the geographical properties of the area surrounding the target vessel, e.g. a channel or a narrow fairway. The incorporation of these two considerations into the current scheme will be pursued in future work.

6. Conclusions

A framework for anticipating the behaviours of ships in multi-vessel encounters was presented in this paper. The proposed scheme comprises a DES-based module for describing current and predicting future intended actions of ships along with a trajectory estimator. The modular architecture of the method allows for flexibility in the design and optimisation of the individual functionalities and guarantees the absence of deadlocks. Dual-view assessment was employed to reduce complexity of the anticipation task and computational load, which would otherwise increase by the square of the number of vessels. The proposed method was tested in simulation environment in three realistic scenarios of multi-vessel encounters with high risk of collision. The results demonstrated that autonomous situation assessment and anticipation of future ship behaviours were achieved at reduced complexity. Future extensions of this work will include consideration of stochastic automata that will incorporate a description for the uncertainty associated to the perception of OS and the tracking of the TVs.

Table 8
Simulation scenario 3: Reports generated by the understanding automata, the coordinator automaton and the associated dual-view assessments. Section 3.1.4 notation is used.

Encounter			Report s_1 from G_u, G_c		$r_1 = P_A(s_1)P_B(s_1)$	$f_{CLR}(r_1)$	r_2	$f_{CLR}(r_2)$
OS	→	TV$_1$	$u_1u_2u_3u_8u_9u_{10}\neg u_{14}c_1c_2c_{13}c_6c_8c_9$	→	$u_1u_2u_3u_8u_9c_{13}$	1	$u_1u_2u_3u_7u_9c_{13}$	0
OS	→	TV$_2$	$u_1u_2u_3u_4u_9u_{10}\neg u_{14}c_1c_2c_{13}c_6c_8c_9$	→	$u_1u_2u_3u_4u_9c_{13}$	0	$u_1u_2u_3u_5u_9c_{13}$	1
TV$_1$	→	TV$_2$	$u_1u_2u_3u_7u_9u_{10}\neg u_{14}c_1c_2c_{13}\neg c_6c_9$	→	$u_1u_2u_3u_7u_9c_{13}$	0	$u_1u_2u_3u_8u_9c_{13}$	1
OS	→	TV$_3$	$\neg u_1c_1c_2c_{13}c_6c_8c_9$	→	$\neg u_1c_{13}$	0	$\neg u_1u_9$	0
TV$_1$	→	TV$_3$	$\neg u_1c_1c_2c_{13}\neg c_6c_9$	→	$\neg u_1c_{13}$	0	$\neg u_1u_9$	0
TV$_2$	→	TV$_3$	$\neg u_1c_1c_2c_{13}\neg c_6c_9$	→	$\neg u_1c_{13}$	0	$\neg u_1u_9$	0
OS	→	TV$_4$	$\neg u_1c_1c_2c_{13}c_6c_8c_9$	→	$\neg u_1c_{13}$	0	$\neg u_1u_9$	0
TV$_1$	→	TV$_4$	$\neg u_1c_1c_2c_{13}\neg c_6c_9$	→	$\neg u_1c_{13}$	0	$\neg u_1u_9$	0
TV$_2$	→	TV$_4$	$\neg u_1c_1c_2c_{13}\neg c_6c_9$	→	$\neg u_1c_{13}$	0	$\neg u_1u_9$	0
TV$_3$	→	TV$_4$	$u_1u_2u_3u_7u_9u_{10}\neg u_{14}c_1c_2c_{13}\neg c_6c_9$	→	$u_1u_2u_3u_7u_9c_{13}$	0	$u_1u_2u_3u_8u_9c_{13}$	1
OS	→	TV$_5$	$\neg u_1c_1c_2c_{13}c_6c_8c_9$	→	$\neg u_1c_{13}$	0	$\neg u_1u_9$	0
TV$_1$	→	TV$_5$	$\neg u_1c_1c_2c_{13}\neg c_6c_9$	→	$\neg u_1c_{13}$	0	$\neg u_1u_9$	0
TV$_2$	→	TV$_5$	$\neg u_1c_1c_2c_{13}\neg c_6c_9$	→	$\neg u_1c_{13}$	0	$\neg u_1u_9$	0
TV$_3$	→	TV$_5$	$\neg u_1c_1c_2c_{13}\neg c_6c_9$	→	$\neg u_1c_{13}$	0	$\neg u_1u_9$	0
TV$_4$	→	TV$_5$	$u_1u_2u_3u_7u_9u_{10}\neg u_{14}c_1c_2c_{13}\neg c_6c_9$	→	$u_1u_2u_3u_7u_9c_{13}$	0	$u_1u_2u_3u_8u_9c_{13}$	1
OS	→	TV$_6$	$u_1u_2u_3u_7u_9u_{10}\neg u_{14}c_1c_2c_{13}c_6c_8c_9$	→	$u_1u_2u_3u_7u_9c_{13}$	0	$u_1u_2u_3u_8u_9c_{13}$	1
TV$_1$	→	TV$_6$	$u_1u_2u_3u_8u_9u_{10}\neg u_{14}c_1c_2c_{13}\neg c_6c_9$	→	$u_1u_2u_3u_8u_9c_{13}$	1	$u_1u_2u_3u_7u_9c_{13}$	0
TV$_2$	→	TV$_6$	$u_1u_2u_3u_7u_9u_{10}\neg u_{14}c_1c_2c_{13}\neg c_6c_9$	→	$u_1u_2u_3u_7u_9c_{13}$	0	$u_1u_2u_3u_8u_9c_{13}$	1
TV$_3$	→	TV$_6$	$\neg u_1c_1c_2c_{13}\neg c_6c_9$	→	$\neg u_1c_{13}$	0	$\neg u_1u_9$	0
TV$_4$	→	TV$_6$	$u_1\neg u_2c_1c_2c_{13}\neg c_6c_9$	→	$u_1\neg u_2c_{13}$	0	$u_1\neg u_2u_9$	0
TV$_5$	→	TV$_6$	$\neg u_1c_1c_2c_{13}\neg c_6c_9$	→	$\neg u_1c_{13}$	0	$\neg u_1u_9$	0

Acknowledgements

The Danish Maritime Foundation, Orients Fund and LauritzenFonden sponsored the first author through the ASAN project (DDMF 2018-056). The ShippingLab WP2 on Autonomy was funded by the Danish Innovations Fund (Grand Solution No. 8090-00063B) and the three private Funds mentioned. The ShippingLab WP2 co-sponsored the first three authors. This support is gratefully acknowledged.

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 $f : X \times E \rightarrow X$ 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-tuple

$$G \triangleq (X, E, f, x_0, X_m) \ . \tag{8}$$

Definition 2 (Event string). A sequence of events $s = e_1e_2 e_3 \ldots$ of arbitrarily large length is called an *event string* or a *word*.

Events and event strings can be concatenated. The *concatenation* of $s_1 = abc$ with $s_2 = def$ is a new string $s_1s_2 = abcdef$. 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 3 (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

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

Definition 4 (Projection). The *projection* of a string s defined over the event set E_1 onto the event set E_2 is defined as

$$P_2(\varepsilon) \triangleq \varepsilon$$

$$P_2(e) \triangleq \begin{cases} e & \text{if } e \in E_2 \\ \varepsilon & \text{if } e \in E_1 - E_2 \end{cases}$$

$$P_2(se) \triangleq P_2(s)P_2(e) \text{ for } s \in E_1^*, \ e \in E_1$$

Definition 5. 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.

B. COLREGS extract

This appendix provides a brief list the COLREGS rules used in the automata. See [22].

Rule 7 - Risk of collision Every vessel shall use all available means appropriate to the prevailing circumstances and conditions to determine if risk of collision exists. If there is any doubt such risk shall be deemed to exist.

Rule 9 - Crossing channel or fairway Any vessel is forbidden to cross a narrow channel or fairway if such crossing impedes the passage of a vessel which can safely navigate only within such channel or fairway.

Rule 13 - Overtaking Any vessel overtaking any other shall keep out of the way of the vessel being overtaken. A vessel shall be deemed to be overtaking when coming up with another vessel from a direction more than 22.5 degrees abaft her beam.

Rule 14 - Head-on situation When two power-driven vessels are meeting on nearly reciprocal courses so as to involve risk for collision, then alter course to starboard so that each pass on the port side of each other.

Rule 15 - Crossing situation When two power-driven vessels are crossing so as to involve risk of collision, the vessel which has the other 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.

Rule 16 - Actions by give-way vessel Take early and substantial action to keep well clear.

Rule 17 - Actions by stand-on vessel Keep course and speed (be predictable) if possible. If it is necessary to take action, then the ship should try to avoid altering course to port for a vessel on her own port side.

References

[1] C. Chauvin, S. Lardjane, G. Morel, J.-P. Clostermann, and B. Langard, "Human and organisational factors in maritime accidents: Analysis of collisions at sea using the hfacs," *Accident Analysis and Prevention*, vol. 59, pp. 26–37, 2013, 99.

[2] EMSA, "Annual Overview of Marine Casualties and Incidents 2014," EMSA, Tech. Rep., 2019.

[3] J. V. Earthy and M. Lützhöft, "Autonomous ships, ICT and safety management," *Managing Maritime Safety*, pp. 141–165, 2018.

[4] P. Agrawal and J. Dolan, "Colregs-compliant target following for an unmanned surface vehicle in dynamic environments," vol. 2015-December, 2015, pp. 1065–1070.

[5] A. Rasouli, I. Kotseruba, and J. Tsotsos, "Pedestrian action anticipation using contextual feature fusion in stacked rnns," 2020.

[6] A. Benterki, M. Boukhnifer, V. Judalet, and C. Maaoui, "Artificial intelligence for vehicle behavior anticipation: Hybrid approach based on maneuver classification and trajectory prediction," *IEEE Access*, vol. 8, pp. 56 992–57 002, 2020.

[7] W. Ding and S. Shen, "Online vehicle trajectory prediction using policy anticipation network and optimization-based context reasoning," vol. 2019-May, 2019, pp. 9610–9616.

[8] J. Talamino and A. Sanfeliu, "Anticipatory kinodynamic motion planner for computing the best path and velocity trajectory in autonomous driving," *Robotics and Autonomous Systems*, vol. 114, pp. 93–105, 2019.

[9] A. Terwilliger, G. Brazil, and X. Liu, "Recurrent flow-guided semantic forecasting," 2019, pp. 1703–1712.

[10] A. Bennajeh, S. Bechikh, L. Ben Said, and S. Aknine, "Anticipation model based on a modified fuzzy logic approach," *IET Intelligent Transport Systems*, vol. 13, no. 2, pp. 260–268, 2019.

[11] N. Wan, C. Zhang, and A. Vahidi, "Probabilistic anticipation and control in autonomous car following," *IEEE Transactions on Control Systems Technology*, vol. 27, no. 1, pp. 30–38, 2019.

[12] R. Ghandour, A. Victorino, A. Charara, and D. Lechner, "Risk indicators anticipation based on the vehicle dynamics anticipation to avoid accidents," 2012, pp. 93–98.

[13] F. Da Cunha, A. Victorino, R. Ghandour, and A. Charara, "Vehicle dynamics prediction based on state observers entries anticipation," vol. 44, no. 1 PART 1, 2011, pp. 2260–2265.

[14] O. Styles, A. Ross, and V. Sanchez, "Forecasting pedestrian trajectory with machine-annotated training data," vol. 2019-June, 2019, pp. 716–721.

[15] B. Murray and L. P. Perera, "A data-driven approach to vessel trajectory prediction for safe autonomous ship operations," in *2018 Thirteenth International Conference on Digital Information Management (ICDIM)*. IEEE, sep 2018.

[16] ——, "An ais-based multiple trajectory prediction approach for collision avoidance in future vessels," vol. 7B-2019, 2019.

[17] ——, "A dual linear autoencoder approach for vessel trajectory prediction using historical ais data," *Ocean Engineering*, vol. 209, 2020.

[18] S. Li, J. Liu, and R. Negenborn, "Distributed coordination for collision avoidance of multiple ships considering ship maneuverability," *Ocean Engineering*, vol. 181, pp. 212–226, 2019.

[19] P. Last, M. Hering-Bertram, and L. Linsen, "Interactive history-based vessel movement prediction," *IEEE Intelligent Systems*, vol. 34, no. 6, pp. 3–13, 2019.

[20] P. Dijt and P. Mettes, "Trajectory prediction network for future anticipation of ships," 2020, pp. 73–81.

[21] T. Li, J. Prieto, and J. Corchado, "Fitting for smoothing: A methodology for continuous-time target track estimation," 2016.

[22] IMO, "*Convention on the International Regulations for Preventing Collisions at Sea, 1972 (COLREGs)*," 2021, http://http://www.imo.org/en/About/Conventions/ListOfConventions/Pages/COLREG.aspx [Accessed: October 19].

[23] D. Kufoalor, E. Wilthil, I. Hagen, E. Brekke, and T. Johansen, "Autonomous colregs-compliant decision making using maritime radar tracking and model predictive control," 2019, pp. 2536–2542.

[24] P. Svec, B. C. Shah, I. R. Bertaska, J. Alvarez, A. J. Sinisterra, K. Von Ellenrieder, M. Dhanak, and S. K. Gupta, "Dynamics-aware target following for an autonomous surface vehicle operating under colregs in civilian traffic," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2013, pp. 3871–3878.

[25] D. Papageorgiou, M. Blanke, M. Lützen, M. Bennedsen, J. Mogensen, and S. Hansen, "Parallel automaton representation of marine crafts' COLREGs-based manoeuvering behaviours," *IFAC-PapersOnLine*, vol. 52, no. 21, pp. 103–110, 2019.

[26] P. Hansen, D. Papageorgiou, M. Blanke, R. Galeazzi, M. Lûtzen, J. Mogensen, M. Bennedsen, and D. Hansen, "Colregs-based situation awareness for marine vessels - a discrete event systems approach," *IFAC-PapersOnLine*, vol. 53, no. 2, pp. 14 501–14 508, 2020, 21st IFAC World Congress.

[27] T. Wang, Q. Wu, J. Zhang, B. Wu, and Y. Wang, "Autonomous decision-making scheme for multi-ship collision avoidance with iterative observation and inference," *Ocean Engineering*, vol. 197, p. 106873, 2020.

[28] K. Dittmann, P. N. Hansen, D. Papageorgiou, S. Jensen, M. Lützen, and M. Blanke, "Autonomous Surface Vessel with Remote Human on the Loop: System Design for STCW Compliance," in *13th IFAC Conference on Control Applications in Marine Systems, Robotics, and Vehicles (CAMS)*. IFAC, 2021, accepted - To appear on Science Direct in IFACpapers OnLine.

[29] K. Dittmann, P. N. Hansen, D. Papageorgiou, and M. Blanke, "Autonomy for Ships: A Sovereign Agents Architecture for Reliability and Safety by Design," in *5th International Conference on Control and Fault-Tolerant Systems (SysTol)*. IEEE, 2021, accepted - To appear on IEEE Xplore.

[30] C. G. Cassandras and S. Lafortune, *Introduction to discrete event systems*. Springer US, 2008.

[31] L. Hu, W. Naeem, E. Rajabally, G. Watson, T. Mills, Z. Bhuiyan, C. Raeburn, I. Salter, and C. Pekcan, “A multiobjective optimization approach for colregs-compliant path planning of autonomous surface vehicles verified on networked bridge simulators,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 3, pp. 1167–1179, 2019.