Human-System Concurrent Task Analysis for Maritime Autonomous Surface Ship Operation and Safety

Ramos, M.[1*]; Thieme, C[1].; Utne, I.[1]; Mosleh, A.[1, 2]

Affiliations

[1] Department of Marine Technology, Norwegian University of Science and Technology Trondheim, Otto Nielsen Vei 10., 7491 Trondheim, Norway

[2]B. John Garrick Institute for the Risk Sciences, University of California. Los Angeles, 404 Westwood Plaza, 90095 Los Angeles, CA, USA

*Corresponding author: Marilia A. Ramos, marilia.a.ramos@ntnu.no

Abstract

Maritime Autonomous Surface Ships (MASS) are the subject of a diversity of projects and some are in testing phase. MASS will probably include operators working in a shore control center (SCC), whose responsibilities may vary from supervision to remote control, according to Level of Autonomy (LoA) of the voyage. Moreover, MASS may operate with a dynamic LoA. The strong reliance on Human-Autonomous System collaboration and the dynamic LoA should be comprised on the analysis of MASS to ensure its safety; and are shortcomings of current methods. This paper presents the Human-System Interaction in Autonomy (H-SIA) method for MASS collision scenarios, and illustrates its application through a case study. H-SIA consists of an Event Sequence Diagram (ESD) and a Concurrent Task Analysis (CoTA). The ESD models the scenario in a high level and consists of events related to all system's agents. The CoTA is a novel method to analyse complex systems. It comprises of Task Analysis of each agent, which are preformed concurrently, and uses specific rules for re-description. The H-SIA method analyses the system as whole, rather than focus on each component separately, allowing identification of dependent tasks between agents and visualization of propagation of failure between the agents' tasks.

Keywords: autonomous ships, autonomous systems, safety, risk

Table of Acronyms

AS	Autonomous ship
AAWA	Advanced Autonomous Waterborne Applications
BBN	Bayesian Belief Network
BP	Branch Point
BUS	Back panel Unit Socket
CC	Collision Candidate
CoTA	Concurrent Task Analysis
ESD	Event Sequence Diagram
H-AS	Human-Autonomous System
HCI	Human-Computer Interaction
HMI	Human Machine Interface
H-SIA	Human-System Interaction
HRA	Human Reliability analysis
HTA	Hierarchal Task Analysis
ID	Identifier
IDA	Information, Decision, Action model
IDAC	Information, Decision and Action on Crew context model
IE	Initiating Event
Lidar	Laser induced detection and ranging
LoA	Level of Autonomy
MASS	Maritime Autonomous Surface Ship
MUNIN	Maritime Unmanned Navigation through Intelligence in Networks
NFAS	Norwegian Forum for Autonomous Ships
PE	Pivotal Event
Radar	Radio aided detection and ranging
REN	Real time Ethernet
SCC	Shore Control Center
STPA	Systems-Theoretic Process Analysis
TA	Task Analysis
TTA	Tabular Task Analysis
TS	Target Ship

1. Introduction

Research and development projects on Maritime Autonomous Surface Ships (MASS) have faced increasing interest, and some are currently in a testing phase. For instance, Yara Birkeland, an autonomous and electric container vessel developed by Yara and Kongsberg, is expected to undergo the first operational tests at the start of 2019, and to conduct fully autonomous operations by 2020 [1]. DNV ReVolt, an unmanned shortsea vessel developed by DNV GL, is being tested in a 1:20 scale, in collaboration with the Norwegian University of Science and Technology (NTNU) [2]. In addition, NTNU is currently testing a 1:2 scaled autonomous passenger ferry, which is expected to run on full scale in 2020 [3]. Several research projects and forums also address MASS, such as the Advanced Autonomous Waterborne Applications Initiative (AAWA) [4]; the Norwegian Forum for Autonomous

Ships (NFAS) [5], [6]; and The Maritime Unmanned Navigation through Intelligence in Network (MUNIN) [7].

The growing number of projects related to MASS is due to the expected advantages it may bring, compared to traditional manned ships. The removal of the accommodation and deckhouse can save cost, weight and space; enable the ship to carry more cargo [4], create more flexible transport solutions [6]; provide better accessibility to potentially dangerous areas [8]; and lead to greener shipping. Moreover, MASS' operation may be safer than traditional manned ships, since human error may be a contributing factor to many marine accidents [9]. Less crew or unmanned ships may also lead to fewer fatalities and injuries if an accident should occur.

Nevertheless, a fully autonomous vessel with no supervision and/or interference from humans is not expected to be a reality in the near future. Most of the projects regarding MASS include operators working in a shore control center (SCC), whose responsibilities may vary from supervision to remote control. MASS operation will thus rely on a human-autonomous system (H-AS) collaboration. Moreover, MASS may have a dynamic Level of Autonomy (LoA): the LoA may change in the same voyage depending on certain conditions. Hence, the operators' tasks may change during a voyage: they may have to control the vessel remotely during parts of the voyage, e.g., when maneuvering in a busy congested harbor, and then change to monitoring the vessel when it moves to the open sea.

As humans will still be involved in the operation at some level, human error may still occur [10]–[12]. In addition to human error, MASS introduce new challenges to the maritime sector, such as increased cyber security threats, the possibility of losing communication with the control center; or the difficulty of performing maintenance during sea voyages [4] [13]. Hence, risk assessments of MASS operation are important [14]. They face two main challenges: i) the strong reliance on H-AS collaboration during the operation, and ii) the possibility of a dynamic LoA.

Few publications address topics related to hazards and risks associated with MASS operation. A recent review [14] of risk models aiming at AS and conventional ships revealed that current models do not sufficiently model the functions carried out by software based systems and that human operators are often treated superficially. Different operational modes of vessels are only covered to a limited extent. The current literature aiming at MASS also does not model and analyse the H-AS interaction as potential contributor to the risk of MASS operation, nor does it reflect the dynamic LoA of the operation. This paper intends to fill this gap, through the development of a method for analysing the operation of unmanned autonomous ships.

The contribution of this article is the Human-System Interaction in Autonomy (H-SIA) method for MASS, which consists of two elements; an Event Sequence Diagram (ESD) and a novel method called Concurrent Task Analysis (CoTA):

i) The ESD models the scenario in a high level and consists of events related to both humans working in a shore control centre (SCC) and the autonomous ship system. We provide a flowchart, in which questions about the design and LoA guide the analyst to build the ESD. The flowchart ensures traceability and reproducibility and is well suited for MASS designed for operations with either low or high autonomy, or for a dynamic LoA;

ii) The CoTA is a new method introduced in this paper to analyse complex systems. It is developed from the ESD. It models the interactions between tasks performed by different agents (e.g., humans and autonomous ships). A CoTA comprises of a Hierarchical Task Analysis (HTA) or Tabular Task Analysis (TTA) of each agent, in which the tasks are re-described until basic tasks that relate to the interaction between the agents are determined. Moreover, the CoTA uses a cognitive model and extends it to the complete system. The CoTA has several purposes, such as failure events identification, failure propagation analysis, and procedures development.

The innovative aspect of the H-SIA method is that the system is analysed as whole, rather than focused on each component separately. In addition, it may be used to compare risks from different MASS designs, or different LoAs during the design phase of the ship, as well as during operation. Further, the CoTA makes it possible to identify dependent tasks between different agents and to visualize propagation of failure between the agents' tasks. Since the CoTA uses the IDA model, it is possible to identify failures related to both humans and the AS.

The paper is organized as follows: section 2 presents the state of the art for autonomous ships risk models and an overview of the background used for the development of the H-SIA method, namely: the ESD, Task Analysis and IDA model. Section 3 presents the H-SIA method, its elements, and its advantages. The method has been developed for collision scenarios, as collision is one of the main causes to ship losses [15]. An application to a potential scenario demonstrates systematically the H-SIA method for MASS, with its strengths and limitations in section 4. Section 5 concludes the paper.

2. Background

2.1 Autonomous Ships and Risk Assessment

Autonomy can be defined as "a system's or sub-system's own ability of integrated sensing, perceiving, analysing, communicating, planning, decision-making, and acting, to achieve its goals as assigned by its human operator(s) through designed human-machine interface (HMI)" [17]. Although the term "autonomous ship" suggests, at a first glance, a concept in which a ship would have a system that is fully responsible for all aspects of its navigation and is independent of humans, autonomous ships may have different LoAs. There are different LoA taxonomies, see, e.g., [27, 28]. In this paper, we use the taxonomy in Table 1 proposed by NFAS [2].

Table 1: Levels of Autonomy for a merchant ship [2]

LoA		Description
1	Direct Control	Direct control of ship, minimal automation and decision support
2	Decision support	Decision support and advice to crew on bridge. Crew decides.
3	Automatic bridge	Automated operation, but under continuous supervision by crew onboard
4	Periodically unmanned	Supervised by shore. Muster crew if necessary
5	Remote control	Unmanned, continuously monitored and direct control from shore.
6	Automatic	Unmanned under automatic control, monitored from shore.
7	Constrained autonomous	Unmanned, partly autonomous, supervised by shore
8	Fully autonomous	Unmanned and without supervision

The white rows of Table 1 relate to the LoAs of manned/ periodically manned ships. Unmanned autonomous ships (grey rows) may operate in four different LoAs, from remote control to fully autonomous. Unless the ship is fully autonomous it has some interaction and/ or collaboration with human operators. The H-AS collaboration will be decisive for safe MASS operations. The crew could be working in a SCC, similar to the MUNIN project [16]. The SCC is a control room in which the operators monitor and supervise unmanned ships or control them remotely, and each operator may be in charge of more than one ship at a time [29].

The Autonomous System comprises, thus the following elements (Figure 1):

i) Operators: human operators working in a SCC;

ii) Autonomous Ship (AS): the autonomous ship including all system on board, such as the software and hardware and communication channels;

iii) Autonomous System: the system comprised of the SCC and the AS – which are named agents of the system.

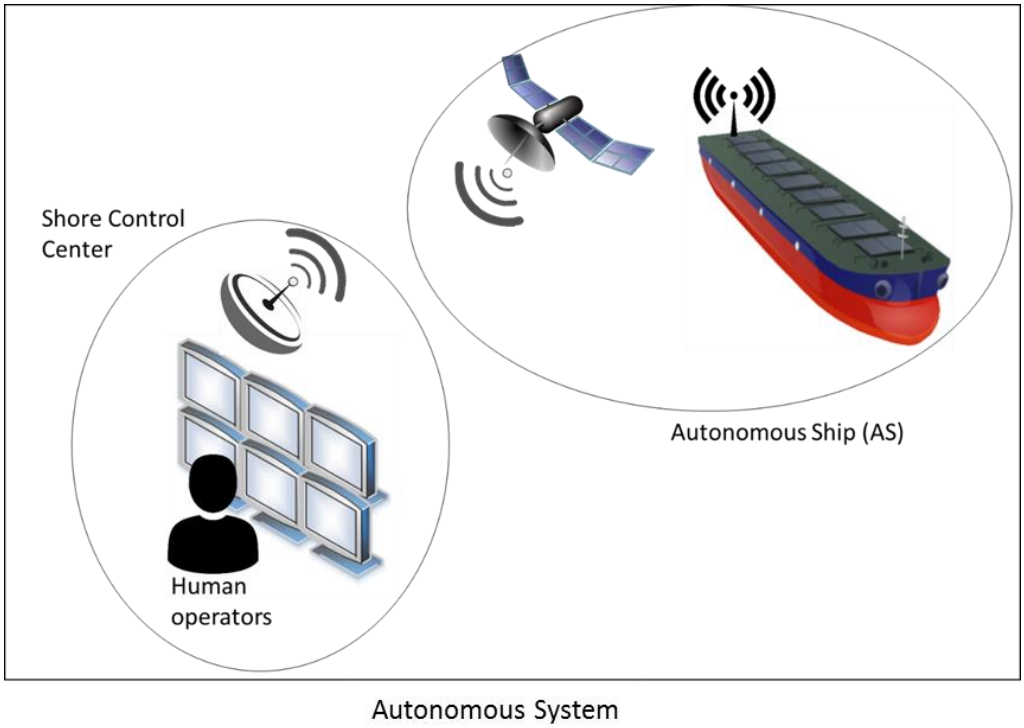


Figure 1: Autonomous System boundaries and elements

Risk assessments for conventional ships have been developed for several decades - recent reviews can be found in Li et al. [17]; Lim et al. [18], or Thieme et al [14]. Since MASS are emerging, there are limited works published on their risk assessments.

Recent publications on risk assessment of MASS derived from the MUNIN or AAWA projects, such as a hazard identification process and risk assessment approach for an early concept of MUNIN [10], [19], and qualitative and quantitative risk assessment of the project case study ship [20], [21], [22]. The AAWA project report [4], highlight issues that need attention with respect to the level of risk of MASS operation.

Publications also include the use of some risk assessment techniques, such as a hazard analysis structured in a Bayesian belief network (BBN) [23]; a what-if analysis to assess the impact of introducing unmanned ships in the maritime industry [24]; STPA and failure mode and effect analysis (FMEA) to identify, analyze and develop verification goals for the DP system of ships [25] [26]; STPA for hazard identification for an autonomous urban ferry project [27], and identification of risk influencing factors (RIF) that impact the risk of MASS[28]. These reviewed publications focus mainly on the identification of hazards. The present article attempts to build risk models for the assessment of the risk level, integrating the complex interactions expected to emerge from MASS.

2.3 Event Sequence Diagram

ESDs can be defined as generalized event trees. The ESD framework allows for indicating the behavior of not only key process variables, but also operator and hardware state changes. It provides thus a more

literal representation of a system state than event trees [29]. An ESD represents the possible sequence of events following an initiating event, which can be a disturbance of the normal state of the system, leading to the possible consequences (end states). ESDs have been applied in several industries and disciplines. ThePhoenix Human Reliability methodology makes use of a flowchart approach to build a Crew Response Tree, which is modelled using ESDs [30]–[32].

An ESD may contain six types of elements [29]: i) events (observable physical phenomenon relevant to the analysis); ii) conditions, which create binary paths; iii) gates, which connect events; iv) process parameter set (time and other parameters that influence the system); v) constraints/ boundaries (set of intervals of process parameters which are in competition with the time to occurrence of an event), and vi) dependency rules, which describe the interaction of the set of process parameters. Figure 2 presents the types of events and gates.

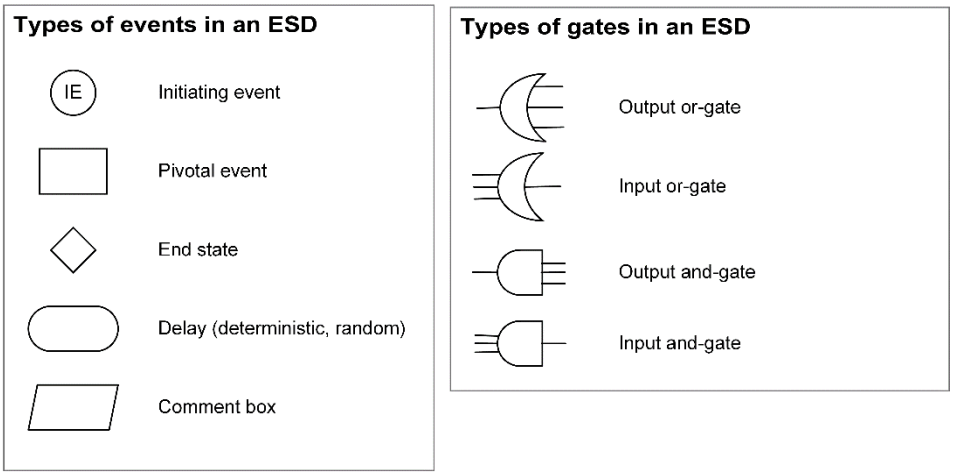


Figure 2: Types of events and gates in an ESD

A strength of the ESD is that it can be used to model dynamic systems [29] and, further, may be used in a dynamic risk analysis of such systems. Moreover, ESDs may be combined with fault tree analysis, BBNs, or a combination of these, in a Hybrid Causal Logic Modeling [33], [34]. Advanced algorithms allow for the quantification of interconnected events in the ESD (ibid, [35])

2.4 Task Analysis

The CoTA developed in the H-SIA method presented in this paper builds upon Task Analysis (TA) theory and methods. TA was developed in the 1960s [36] and had the initial focus of analyzing human performance. TA has since developed, influenced by the technical challenges in the Human-Computer Interaction (HCI) [37]. Diaper [38] suggested a definition for using TA in HCI, which follows a systems perspective rather than emphasizing human performance only. Task analysis is “the collective noun used in the field of ergonomics, which includes HCI, for all the methods of collecting, classifying, and interpreting data on the performance of systems that include at least one person as a system component”.

A TA can be developed through different approaches, such as HTA, TTA, and Cognitive Task Analysis. More information on the different approaches can be seen in [37].

HTA offers the possibility of analyzing complex tasks through the decomposition of goals into sub-goals – a process named re-description. The goals and sub-goals are organized in HTA through *plans* [36]. Plans state the order in which the sub-goals must be accomplished. From a systems perspective, the HTA should focus on the analysis of the task to understand how the system is supposed to behave and, further, how it can fail. HTA is a well known tool in the field of Human Reliability Analysis (HRA) and the second step in a general HRA process [39].

The flexibility of the plans of HTA and its hierarchical structure allows the modeling of the expected behavior of a diversity of parts of the system. Moreover, the re-description of the goals into sub-goals allows the identification of specific tasks in the desired component level. For instance, since a task in software engineering is a computational operation that can be executed concurrently with other computation tasks [40], it is possible to apply HTA in software reliability analysis. A task is thus a function or a function object of a software program. Indeed, TA is similar to a functional decomposition that is often carried out during software development.

An important element of HTA is the stop rule, which determines when a re-description should end. Without an appropriate stop rule, re-description may continue indefinitely. A commonly used subjective stop rule is: stop when you have all the information you need to meet the purposes of the analysis [38], i.e., the task analysis will be detailed until the point that is necessary for the analysis. A HTA for operators supervising/ remotely controlling autonomous ships was developed by [12]. It focuses only on human actions and not tasks of the autonomous ship. It provides a valuable insight into human tasks in a SCC, and it makes use of the operator cognitive model IDA. In this work we expanded IDA to use it as a stop-rule for both operators and AS. The IDA model is described in the next sub-section.

2.5 Information-Decision-Action (IDA) model

IDA – Information, Decision and Action was initially developed as a human behavior model of the response of a nuclear power plant crew under accident conditions [41]. It consists of the cognitive phases I (Information collection and pre-processing); D (decision making and situation assessment); and A (action taking). Since its first publication it has been the subject of diverse developments, such as the development of a crew-centered version named IDAC (Information, Decision and Action in a Crew context) [25], [50]–[53].

It is possible to extend IDA and adapt it for different agents of a system. This is particularly useful when analyzing the interaction between two or more agents, as it allows for decomposing functions into the same low-level unit of analysis. Figure 3 presents the elements of the IDA model in an extended version, generalized for an autonomous ship modeling, in addition to operator behavior, described below.

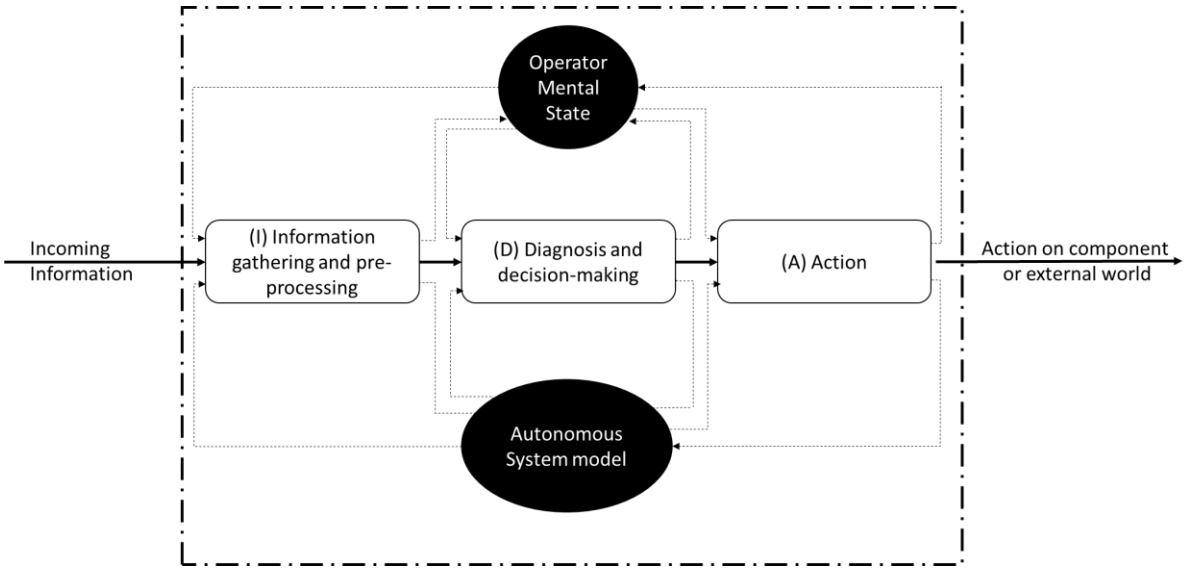


Figure 3: IDA model extended for operator and autonomous ship

In the IDA model the agent receives information from the external world. This may be an alarm, in case of an operator, or data, for the AS system. This information is received and processed through the (I) block, which includes filtering, comprehension, grouping and prioritization. The next phase, (D), relates to the agent's response to the information obtained in the (I) phase. This covers situations assessment, diagnosis and response planning. For the operator, the cognitive response to information obtained in the previous phase is translated into a problem statement or a goal, requiring resolution. The process of problem-solving or goal-resolution involves the selection of a problem-solving method or strategy. These strategies are also found in autonomous systems. The decisions from the (D) phase are put into action through the (A) phase. Typically, the operator performs an action on an external world, whereas a component of the system may perform an action on another component or on the external world.

The I-D-A process is influenced by a "mental" state, for the operators, or an AS model, for the autonomous ships. For the operator, the mental state combined with memory represents the operator's cognitive and psychological states. It explains why and how a response process initiates, why and how a cognitive activity starts and continues, and why and how a goal or strategy is selected or abandoned. For a system, in this case the autonomous ship, the AS model includes the programmed behavior, the process models in the AS, and the world model of the AS.

The interaction between the mental state of the AS model and the I-D-A activities is a dynamic process of mutual influence: the mental state and the AS model influence the activities within each of the IDA blocks, and as a result of these activities the mental state and the AS model are updated (dashed lines in Figure 3).

The goals analyzed in a HTA represent the needs of the system as whole, e.g., operator shall respond to an alarm, or AS shall receive an input from the operator. This is an external reference point in IDA. A mismatch between the agents' actions and this requirement would then be classified as an error, i.e., failure to respond to an alarm, or failure in receiving input. Re-describing the goals as one of the IDA stages (e.g., listening to an alarm is a sub-goal in the information gathering stage) allows then for identifying errors with respect to the internal reference points.

In terms of errors, IDA recognizes errors and failures attributed to humans, hardware, or software as only recognizable in the context. For instance, closing a valve might be an error in one context, but success in another [51]. Thus, errors are defined with respect to external reference points: the needs of the system. The errors identified by the external reference points can be traced to the stage where the error originated: I, D, or A. An error in one of the I-D-A phases means that the error has occurred in the module for which there was correct input but incorrect output. For example, an action execution error is when the action was incorrectly provided with the correct decision. On the other hand, if the action was performed based on a wrong decision, it is concluded to be a decision error.

The advantage of identifying the errors within one of the I-D-A phases is that, for the operator, it can be further traced to the cognitive process leading to the error; and for the system, it can be traced to the responsible component in which the error occurred (e.g., an error in information retrieval can be traced to the responsible sensor). Moreover, performing HTAs in parallel for two agents, as in the CoTA, ensures that low-level tasks from all agents are in the same level of re-description.

3 H-SIA method for MASS collision scenarios

The H-SIA method, presented in this Section, is composed of two elements: (i) an ESD (Section 3.1), and (ii) a CoTA (Section 3.2). The method is specifically developed for collision scenarios between an autonomous ship and another vessel or object, but is nevertheless expected to have general applicability for autonomous systems.

Error! Reference source not found. presents the three main steps in the H-SIA method, and Steps 2 and 3 are detailed in the following sub-sections. The first step is familiarization with the system and its LoA. This ensures that the analyst can use the flowchart for the ESD development (Step 2), in which the questions are related to design of the AS and the LoA of its operation. Following the development of the ESD, the analyst can continue for the CoTA development (Step 3), which builds on the events of the ESD.

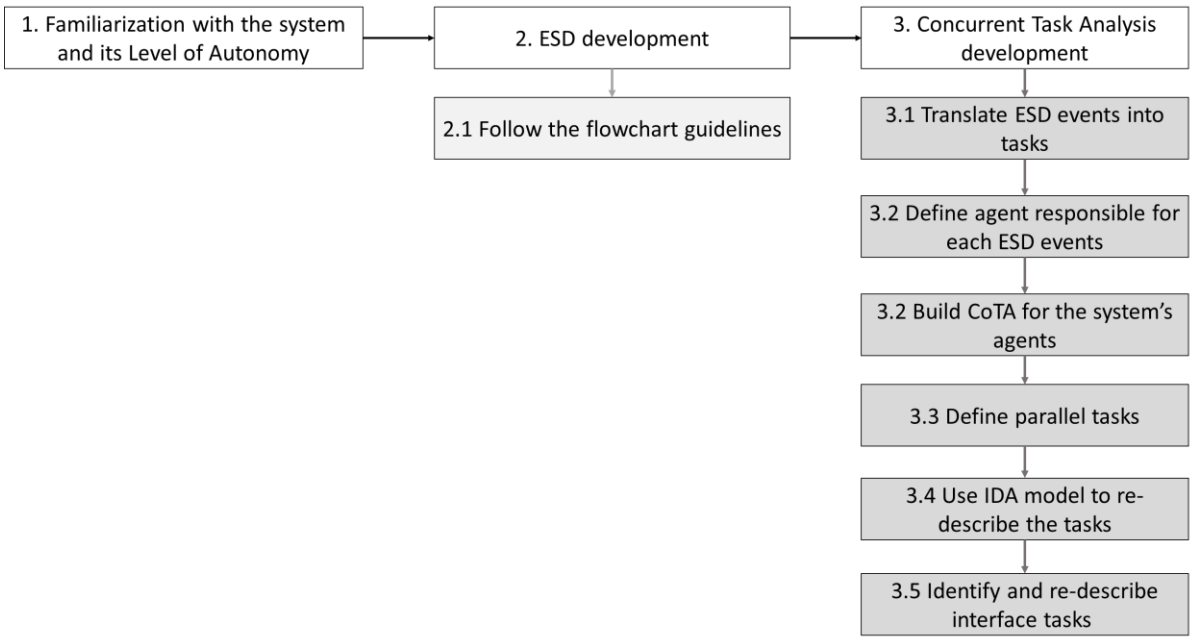


Figure 4: H-SIA method application steps

Figure 5**Error! Reference source not found.** presents a simplified example of the method approach. The CoTA is success-oriented; it describes the tasks involved in the success paths of the events of the ESD. Following the tasks re-description using the CoTA stop rules, the interaction between the interface tasks of the agents are indicated (circles in Figure 5). Note that the events in the ESD cover both events from the AS and from operators working in a SCC. The AS, in this case, includes the autonomous ships and all systems on board, i.e., software and hardware parts, as illustrated in Figure 1.

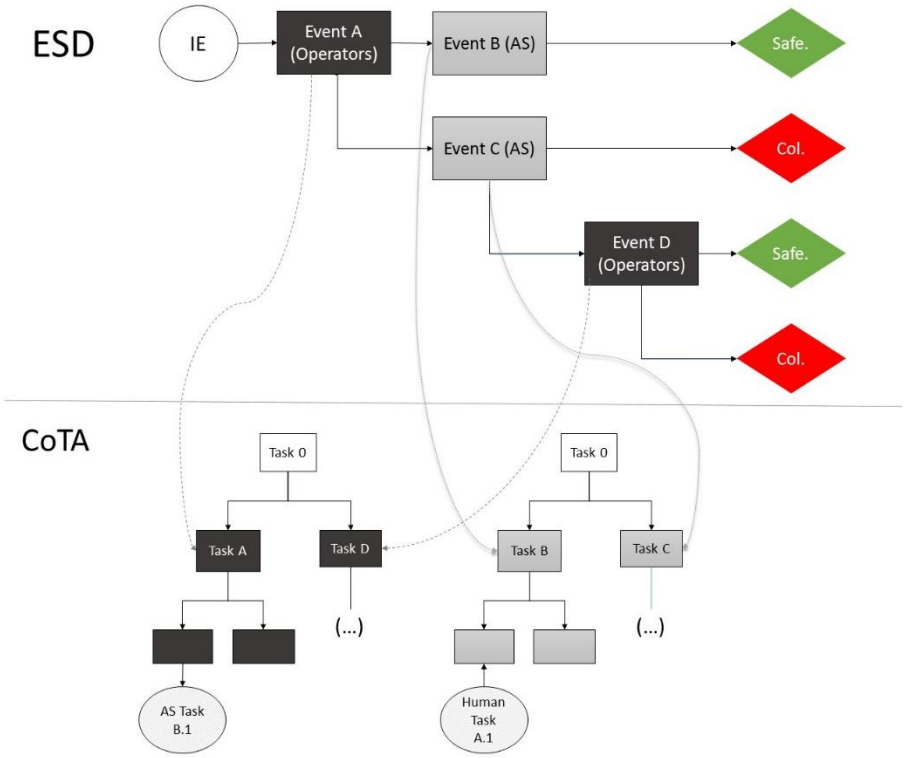


Figure 5: Simplified example of H-SIA method elements

3.1 Event Sequence Diagram

Since the H-SIA method is focused on ship collision scenarios, it is important to understand how the process of collision avoidance can take place for autonomous ships.

3.1.1 Collision Scenario for Autonomous Ships

Collision refers to contact between two or more vessels, or between a vessel and an object, which can be floating or a fixed structure. The types of collisions considered in this paper for Autonomous Ships can be seen in Figure 6. There is no formal definition of a ship being on collision course. Often it is stated that if no course is changed, the vessels will collide [17], [42]. The AS will collide with the collision candidate if the course / speed of the vessel(s) is not changed;

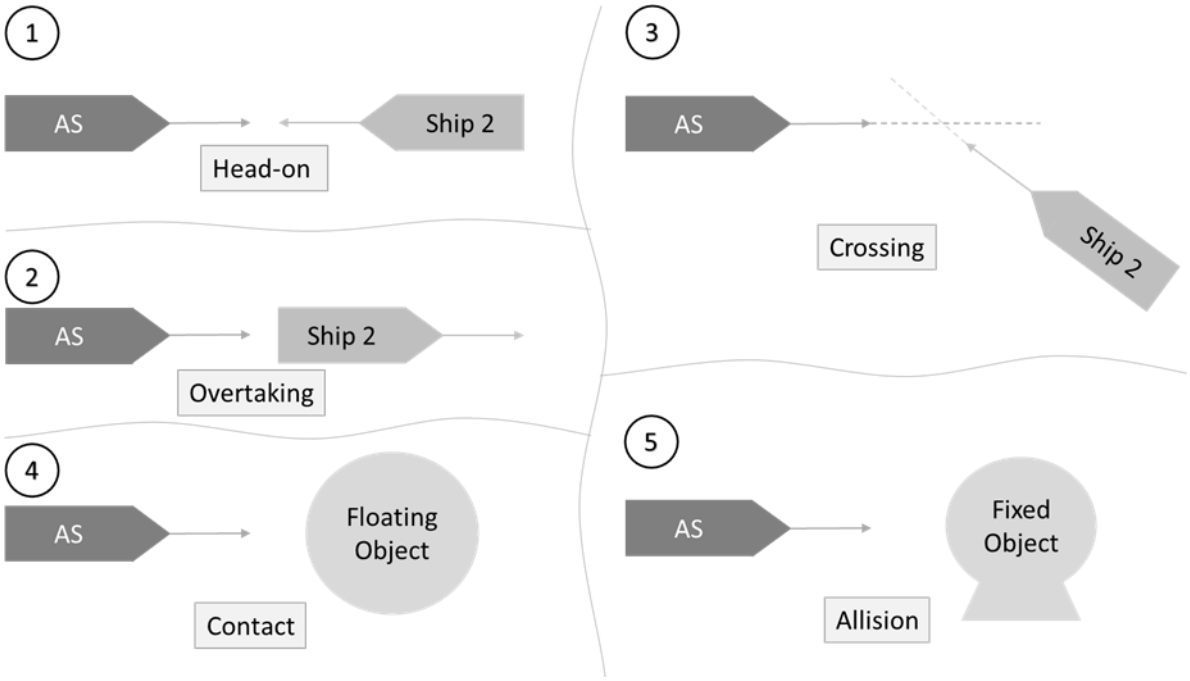


Figure 6: Types of ship accidents related to collisions

Ships at sea must follow the international rules for collision avoidance – COLREGs [42] and local rules. If there are local rules, they have to be considered over the COLREGs. This may involve changing the heading or the speed depending on the traffic situation, environmental conditions, distance from the vessel/object, and ship condition variables, such as, maneuverability of the ship. General rules include to keep an appropriate lookout, and a safe speed for the situation, even if no collision is immanent. For the scenarios in Figure 6 the following rules apply. For head-on scenarios (1), both powered vessels need to change their course to starboard (right in the direction of travel, Rule 14). For scenario 2, the overtaking vessel, in this case the AS is responsible for avoiding collision (Rule 13). Crossing collision require the ship having the crossing ship on starboard to alter course and speed to avoid collision (Rule 15). In this case, it is assumed that the AS has to make a collision avoidance maneuver. Similarly for scenarios 4 and 5, the AS has to take actions to avoid collision with objects. Note that for scenario 4 the floating object may also be a drifting vessel or a fishing boat. These cannot avoid collision due to their state and the AS needs to take actions.

The detection of a collision candidate, the decision making to avoid collision, and the action itself are, in traditional manned ships, responsibilities of the bridge crew. For autonomous ships, the main responsible for these tasks may be either the operator or the ship itself, depending on the LoA. For a ship in remote control (low LoA), the detection may be a task of the operators, and they are responsible for the decision making and for sending the command to the AS. On a fully autonomous system (high LoA), on the other hand, the three tasks would be a responsibility solely of the system. Moreover, it is possible that one agent is the main responsible for a specific task and, if failed, the other agent can act

as recovery. For instance, in a constrained autonomous mode (cf. Table 1), the development of a collision avoidance plan is the responsibility of the AS. The operator may, however, perceive that the plan is inadequate, and take remotely control of the ship. S/he will thus be responsible for the plan and send commands to the AS. The high-level tasks for low and high LoAs are illustrated in Figure 7.

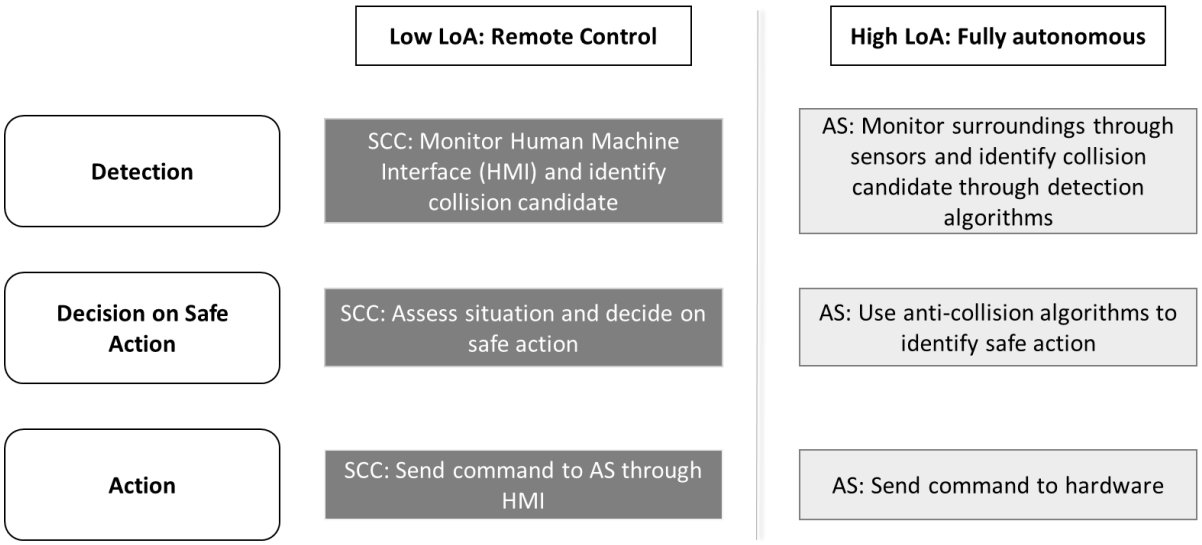


Figure 7: High level tasks for low and high Levels of Autonomy

3.1.2 Assumptions in the ESD flowchart

To facilitate and guide the analyst in the development of the ESD, a flowchart has been developed with the following assumptions:

1. Manning: The autonomous ship is unmanned during all phases of the voyage;
2. Initiating event: The initiating event of the ESD is the autonomous ship being on collision course.
3. The CC can be an autonomous ship, a conventional ship, a fixed object, or a floating object;
4. Communication: If the Collision Candidate is a vessel, it is possible for the crew on the SCC to establish communication with it, and for that vessel to demand communication with the SCC crew;
5. Conservative approach: The AS is the main responsible for avoiding collision, as is depicted and explained in the scenarios[1]. If the CC is a ship and not a fixed or floating obstacle it is possible that it:

[1] According to Rule 17 of COLREGs, if the vessel who has not to take action detects that the collision avoidance by the giving way vessel is insufficient, it shall take actions to avoid collision. However, we adopt a conservative approach in which the CC would only take these actions if contacted by the SCC, even when facing Rule 17.

    a. Takes actions that puts it back on collision course with the AS even after the AS took actions to avoid it, and
    b. Takes actions to avoid collision only after being contacted by the AS;
6. Alarm: In case of detection of a CC by the AS, it can send a sonorous and/or visual alarm to the SCC;
7. SCC crew: the SCC crew is composed by teams. An operator is responsible for monitoring more than one AS at a time, and an experienced supervisor is available;
8. End states: the possible end states are collision, no collision and safety. "Safety" concerns the case when the collision has been avoided and the system is fully functional. "No collision", on the other hand, refers to a situation when collision has been avoided but the AS has a problem regarding its manoeuvrability or control. In this case, the ship will not collide but will also not be operating safely. Although in a first glance "no collision" and "safety" could be merged into one end state, the differentiation is important to stress that, if reaching "no collision", the ship would still need further attention and/ or repair to reach a safe state.
9. If the SCC loses control of the AS, they can try to communicate with the CC, if it is a vessel. Then that vessel can take measures to avoid collision. Since the AS is not controllable or has limited maneuverability, the outcome, in this case, would be "no collision".

3.1.3 Event Sequence Diagram flowchart

To aid the analyst, the ESD is constructed by following the flowchart in Figure 8. The flowchart ensures traceability and reproducibility of the analysis, in addition to aiding the analyst to consider all relevant events, including recovery actions. The development of the flowchart has evolved from analysis of several likely collision scenarios for autonomous ships, considering the potential interaction between operators and AS. This has then been generalized to cover all possible LoAs in Table 1, within the assumptions presented in the previous sub-section.

The flowchart is composed of questions and Branch Points (BPs). The questions are related to the LoA and the design of the system. Depending on the answers of the questions, the BPs will then be present in the ESD as pivotal events. Note that the possible outcomes of the events may be not only failure and success, but also different types of operation (manual or autonomous). The elements for using the flowchart are i) guidelines (presented in the flowchart in Figure 8); ii) questions (Table 2), and iii) BPs description (Table 3). The use of the flowchart may be simplified, in the future, with a software tool, in which the analyst would answer the questions and be provided with the final ESD.

Table 2: ESD Flowchart Questions

Number	Question
1	Who is primarily responsible for the detection of CC?
2	Can the operators detect the CC from the SCC?
3	Can the operators remotely control the AS from the SCC?
4	Can the operator use other measures to avoid collision, c.f., BP J?
5	Who is primary responsible for developing the collision avoidance plan?
6	Is there an alarm in the SCC warning about the CC, and are they informed on the plan for collision avoidance? **Note: If there is no monitoring from a SCC, the answer is NO**
7	Is the collision candidate a ship?
8	Is there enough time available to re-plan and implement a new plan to avoid collision?

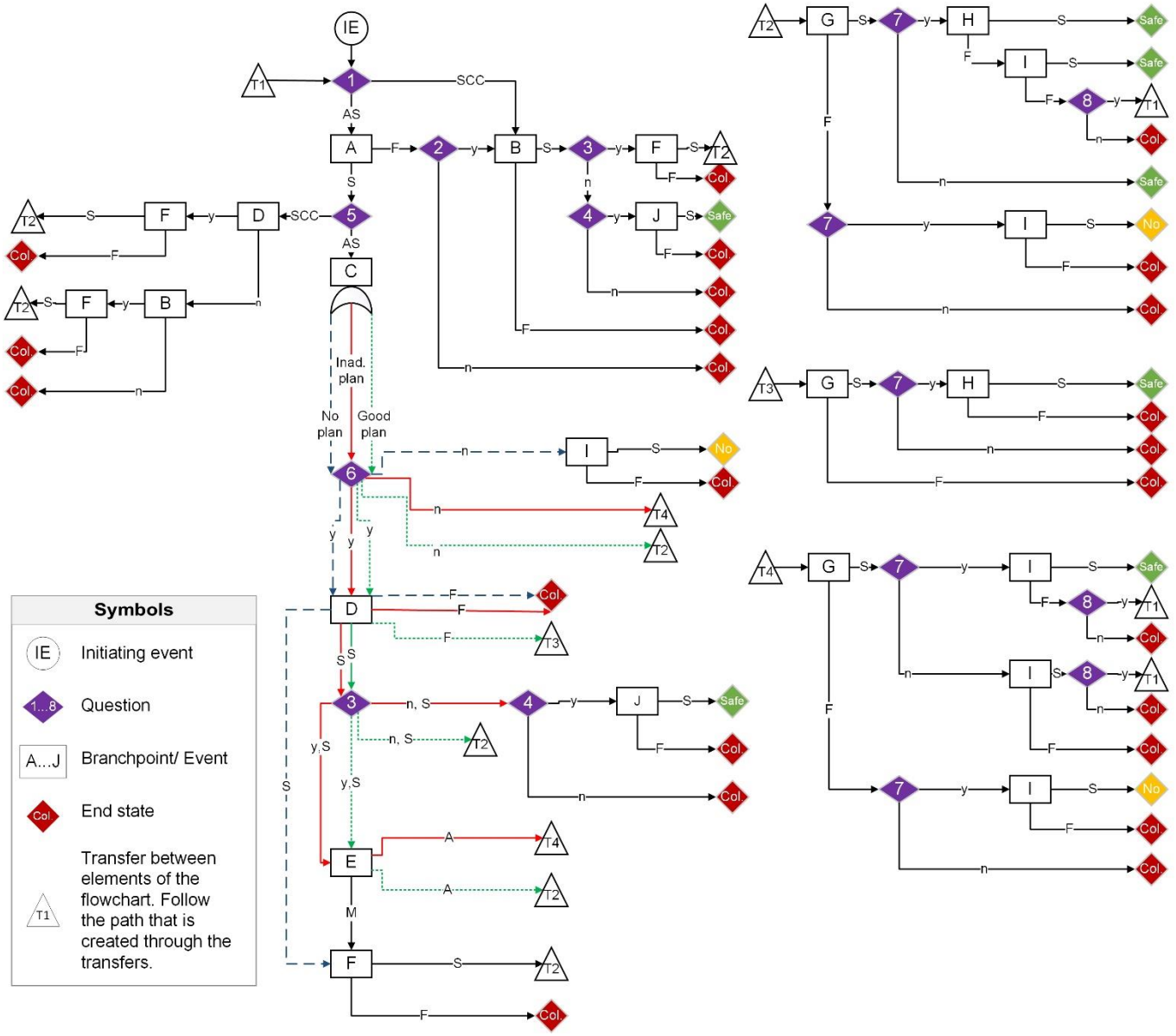
Table 3 presents the description of each branch point, possibly involved subsystems, and its potential outcomes. The subsystems are the components that are involved in the pivotal events. For instance, for the event “detection of the collision candidate by the AS” the following components are necessary: sensors for object detection; computer systems for analysis of data and detection of objects and candidates; antennas and receivers; communication channels/ data network . Failure or success of this event depends, therefore, on the well-functioning of the involved subsystems.

Table 3: ESD Flowchart Branch Points and corresponding events description. IE is initiating event. PE is pivotal event.

BP	**Name (Event)**	**Type**	**Description**	**Involved subsystems**	**Outcomes** *(SP: Success Path; FP: Failure Path)*	
	AS on collision course	IE	The AS will collide with the collision candidate if the course/ speed of the vessel(s) is not changed. AS has to perform actions to avoid collision according to COLREG or local rules.			
A	Detection of CC by AS	PE	The AS has to detect the CC autonomously. The AS shall make use of available sensor systems (e.g., RADAR; LIDAR; camera, AIS) for data collection. Computer algorithms use and combine the data to identify the CC as a collision candidate.	Sensors necessary for object detection; Computer systems for analysis of data and detection of objects and candidates; Antennas and receivers; Communication channels/ data network of equipment (BUS, REN)	**SP:** The AS successfully detects the CC, with correct position, course and other relevant information that might be necessary.	**FP:** The AS does not detect the CC, *or* does not detect the CC as collision candidate, *or* wrongly categorizes CC.
B	Detection of CC by the operators	PE	The detection of the CC is done by the operators who are monitoring the AS from the SCC. This can happen if the operators are the main responsible for detection, or if the detection is not done successfully by the AS. The operators might detect the CC earlier as collision candidate than the AS.	Data availability, sensor systems (camera, radar, LIDAR) communication between ship and SCC HMI and associated equipment Operator performance	**SP:** The operator successfully detects the CC as collision candidate.	**FP:** The operator does not identify the CC, or does not identify the CC as collision candidate.
C	AS generates plan(s) for collision avoidance	ESD Or gate	The AS produces and evaluates different plans for collision avoidance, choosing the optimal maneuver. This may be changing route, speed, and other variables. The AS shall adhere to	Computer system associated with path planning and evaluation Collected data quality and reliability	**SP:** A plan is generated that avoids collision, without creating additional hazardous situations, considering all regulations,	**FPs:** There are two possible failure paths of this event: **FP 1:** The AS does not provide a plan. **FP 2:** The AS provides a plan, which is not adequate for the situation or

BP	Name (Event)	Type	Description	Involved subsystems	Outcomes *(SP: Success Path; FP: Failure Path)*	
			COLREG or local rules and consider the circumstances in the current region.		environmental and regional characteristics.	adhering to the local/ international rules.
D	Operator responds to alarm	PE	The AS sends the collision avoidance plan to the SCC. An alarm warns the operator about the potential collision. The operator has to visualize/ hear the alarm and identify the AS of concern and causes to the alarm. If the operators do not respond to the alarm it is considered that they will not take any action during the whole chain of events.	Communication system between AS and SCC Human operator HMI and computer system	**SP:** The operator successfully responds to the alarm.	**FP:** The operator does not see or hear the alarm. Or the operator fails to assess the situation correctly, being not able to identify the correct AS, or reason for alarm.
E	Decision on operational mode	PE	The operator is aware of the CC and the plan of the AS to avoid collision. The operator has to assess the plan. If it is not adequate, the operator has to take remote control. The outcomes of this BP are not success or failure, but Autonomous or Manual/ remote control.	Communication system between AS and SCC Human operator HMI and computer system Data storage and retrieval of AS	**Manual Operation** The operator decides to take remote control of the AS	**Autonomous Operation** The operator decides for autonomous operation
F	Remote control of the AS	PE	Having decided to remotely control the AS, the operator has to take remote control. This includes sending updated route information or taking direct control through the HMI of the AS.	Human operator – planning and sending HMI and computer system	**SP:** The operator successfully plans and sends a strategy that will avoid collision and not lead to any hazardous situation.	**FP:** The operator fails to send an updated strategy. The operator does not send the strategy in time. Or the sent strategy is insufficient or incorrect, thus leading to a new hazardous situation.
G	Implementation of collision avoidance plan	PE	The AS receives the plan from the operator / the navigational module and implements it. The implementation comprises two phases, determination of necessary control input and sending the control input to the required equipment. If AS does not receive a plan from the SCC, it will implement its own plan.	Communication AS-SCC Communication of AS internally Control software and computer Rudder/ propeller, engine and associated hardware equipment	**SP:** The plan is translated correctly to control input and successfully executed.	**FP:** The AS fails to receive the plan from the operator. Or it fails to produce control input. It fails to change set the system to the course. Or it implements the maneuver differently from the plan.

BP	Name (Event)	Type	Description	Involved subsystems	Outcomes *(SP: Success Path; FP: Failure Path)*	
H	CC follows the rules (if it is a ship)	PE	If the CC is a ship, it can execute its own maneuvers. If these are according to traffic rules, it will not interfere with the scenario. Important: it is considered that the TS is not initiating a recovery action *(it would only do a recovery action if warned by operators from SCC – next BP*)	-	**SP:** CC behaves as expected.	**FP:** CC fails to adhere to rules or executes maneuvers that set it on collision course.
I	Monitoring through the AS and SCC, and CC takes recovering actions.	PE	The AS and operators assess and identify if the AS maneuvers are sufficient to avoid collision. This requires monitoring the implementation of the plan, way progress of AS and TS. In case of actions being not sufficient, the TS needs to be warned and make avoidance move – if the CC is a ship. TS might also consider taking contact with AS and consequently SCC. It is considered that at that moment it is not possible for the AS to correct path or actions, but the TS may act to avoid collision.	Communication equipment Sensors Operators performance TS responses Control system Data collection and evaluation system. HMI in SCC and computer systems AS internal communication	**SP:** SA or operators successfully monitor the situation and if necessary warn the TS successfully. The TS then successfully avoids a collision.	**FP:** The SA and/or operators fail to monitor and/or notice unsuccessful maneuvers. They fail to warn the TS and the TS fails to recognize hazardous situation and communicate with SCC. The TS fails to maneuver to avoid collision. The Failure path may lead to safety or to a "no collision" status. The latter arises from the situation where AS is out of control for lack of communication, system failure, etc. (failure path of H).
J	Other measures the SCC can take	PE	The SCC can take other measures to avoid collision, which do not include controlling AS or the TS taking collision avoidance measures. This may be an emergency shutdown system, deflection by calling on another ship, or sending a rescue ship. This BP allows the analyst to include events that were not accounted for in the previous BPs.	Depend on the scenario.	**SP:** The measure successfully avoids collision.	**FP:** The measure does not avoid collision.



Symbols

IE	Initiating event
1...8	Question
A...J	Branchpoint/ Event
Col	End state
T1	Transfer between elements of the flowchart. Follow the path that is created through the transfers.

Flowchart Guidelines

The numbers (1-8) in the flowchart are questions.

The possible answers are yes/no, SCC/ Autonomous Ship. Refer to the questions table to answer it.

The letters (A-J) indicate an event (branch point).

Each event has a failure and success resulting path; or multiple paths if connected to an OR gate. Refer to the Branch Points Table for the description of the events and its possible existing paths

The answers of the questions define if a branch point will exist or not.

1. Start at the initiating event (IE)
2. Answer question 1
3. Follow the line related to the answer (SCC / Autonomous Ship)
4. Connect the following event to the initiating event
5. Follow the success AND the failure paths of the event (until all consequences are assigned)
6. Answer the questions that follows on a success or failure path
7. Only the path exiting the question that is related to a correct answer will exist on the ESD
8. Note that if the answer to question 5 is "Autonomous Ship", you must follow all paths exiting the "Or"-gate
9. Note that some paths lead to a transfer gate. Follow the direction of the arrow and continue with the flowchart.

Abbreviations

A	Autonomous	n	No
AS	Autonomous Ship	S	Success
F	Failure	SCC	Shore control center
M	Manual (remote) control	y	Yes

Figure 8: ESD Flowchart. Cf. Table 2 for questions.

The use of the flowchart in Figure 8 can be illustrated by the following example: a scenario with a low LoA, in which the operators are the main responsible for detection of the CC and can remotely control the AS. Starting at the IE, the answer to Question 1 is “SCC”, which determines that the next BP will be “B” in the Figure. The analyst thus does not follow the path of “AS” from question 1. From BP B, the analyst follows the Success and Failure paths. The Failure path leads to collision, and the Success path leads to Question 3. The answer to question 3 turns out to be “yes”, which leads to BP F. The Failure path from F leads to collision, and the Success path to gate T2. The analyst then continues to BP “G”, which is the first one after T2, and from there to Question 7. Figure 9 shows the flowchart for this scenario, in which the red lines are the ones followed by the analyst and the dashed ones are to be ignored; and the resulting ESD (to be continued after gate T2 – using BP “G”).

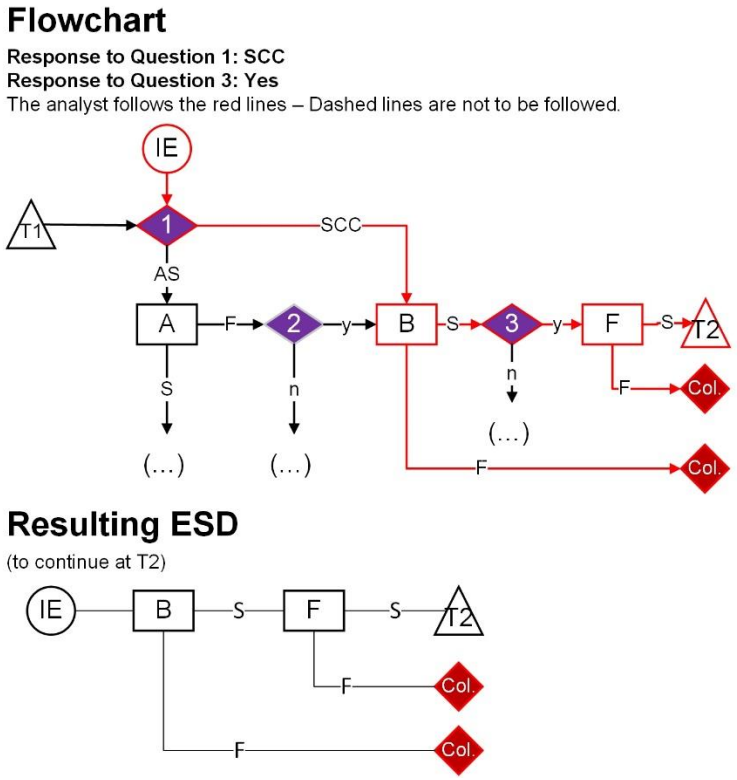


Figure 9: ESD flowchart use example

The purpose of the questions is to include only relevant issues in the ESD that appear in the logic order of the questions. If the SCC, for instance, is the only responsible for detection, the event of the AS having success or failure in detecting the CC is not relevant for the analysis of this system and this AS operation.

3.2 Concurrent Task Analysis

CoTA is a new method specifically developed for the H-SIA method. It is a multi-purpose method for complex system analysis that makes use of TA. It includes the interactions between different parts or agents of the systems.

A CoTA comprises of several HTAs or TTAs, in which the tasks are re-described until basic tasks that relate to the interaction between the agents are found. The CoTA has, thus, specific stop-rules. In addition, the CoTA includes a new type of task named "parallel task". Parallel tasks are supporting tasks, i.e., they are not directly related to the events in the ESD, but are necessary for the execution of the other tasks and the interaction between the agents. Moreover, they are related to the normal operation of the system and should be executed during the whole operation, not following a specific order in a plan, i.e., at the same time of the other tasks. The parallel tasks are normally the ones related to gathering data, monitoring, or communication between the agents.

The CoTA is developed from the system's ESD (step 2). In this process, the events from the ESD are translated into tasks to be performed by the agents. Hence, the ESD presents *what* can happen, and the CoTA further details *how* these events come to place. The CoTA is a success-oriented method that allows for a more detailed understanding of the tasks each agents of the system must accomplish for the events of the ESD to take place.

CoTA can be used to; i) analyze the tasks involved in all events of the ESD, or to ii) analyze a specific sequence of events in the ESD scenario. When developed for all the events of the ESD (alternative i), the CoTA provides a detailed overview of how the agents should behave to be successful in the events of the ESD. It does not, however, provide an overall sequence of how these tasks should be performed; since this is specific for each path (scenario) of the ESD. The scenario-specific CoTA (alternative ii), on the other hand, presents the tasks that should be performed for a success outcome in a specific sequence of events.

The CoTA can be used for multiple purposes, such as to:

i) develop procedures

The CoTA details tasks that must be performed by the agents of the system and provides plans on how to perform those in order to achieve a goal. It can thus be used to develop procedures for operators working in a SCC, or for other parts of the system.

ii) identify the specific subsystem/ component for success of the task

The tasks of the CoTA are re-described until they only fit into one phase of the I-D-A model. The breaking down of a large task into smaller "units" allows for tracing the responsible components for the success of that main task.

iii) identify possible failures from Human and the autonomous ship

Because the CoTA uses the IDA model, it is possible to identify possible failures from the humans and autonomous ship. Errors in IDA are defined according to external reference points: the autonomous system's needs described as the HTAs tasks. A failure on accomplishing them will be then a human or technical failure. The Failure events are defined for low level tasks, and include not performing the task, performing it inadequately, and performing it too early/ too late. It comprises thus errors of omission and commission.

iv) identify tasks to be achieved for reaching a specific outcome

The CoTA may either contain all events from the ESD or be scenario-specific. In either case, the analysis shows which tasks may lead to the safe end states of the ESD.

v) Identify interface tasks

The stop rule of the CoTA is the re-description of the tasks until being related to only one of the IDA phases and, further; assure that the interface tasks – the ones that need an input or output of the other agent – are described. This makes it possible to identify dependent tasks, for which the success will not depend solely on only one of the agents, but on all agents involved in the task. For instance, the response to an alarm depends on the cognitive I phase of the operator, but also on the input sent from the AS to activate the alarm. Although the dependency may seem obvious for some simple tasks, it may be more complex for others. In particular if the analysis comprises several tasks and / or more than two agents. The CoTA provides a process for identifying these dependent tasks and for visually representing them.

vi) Analyze propagation of failure

The identification of dependent tasks enables analysis of failure propagation between the agents.If a task of agent *x* needs an input from a task from agent *y*, then the failure of the latter will cause the failure of *x*. It also helps identifying how one agent's task can be a recovery mechanism from a task failure of another agent. Table 4 presents an example of the identification of propagation of failures and recovery

tasks, which also concern the case study presented in this paper; please refer to the HTAs at

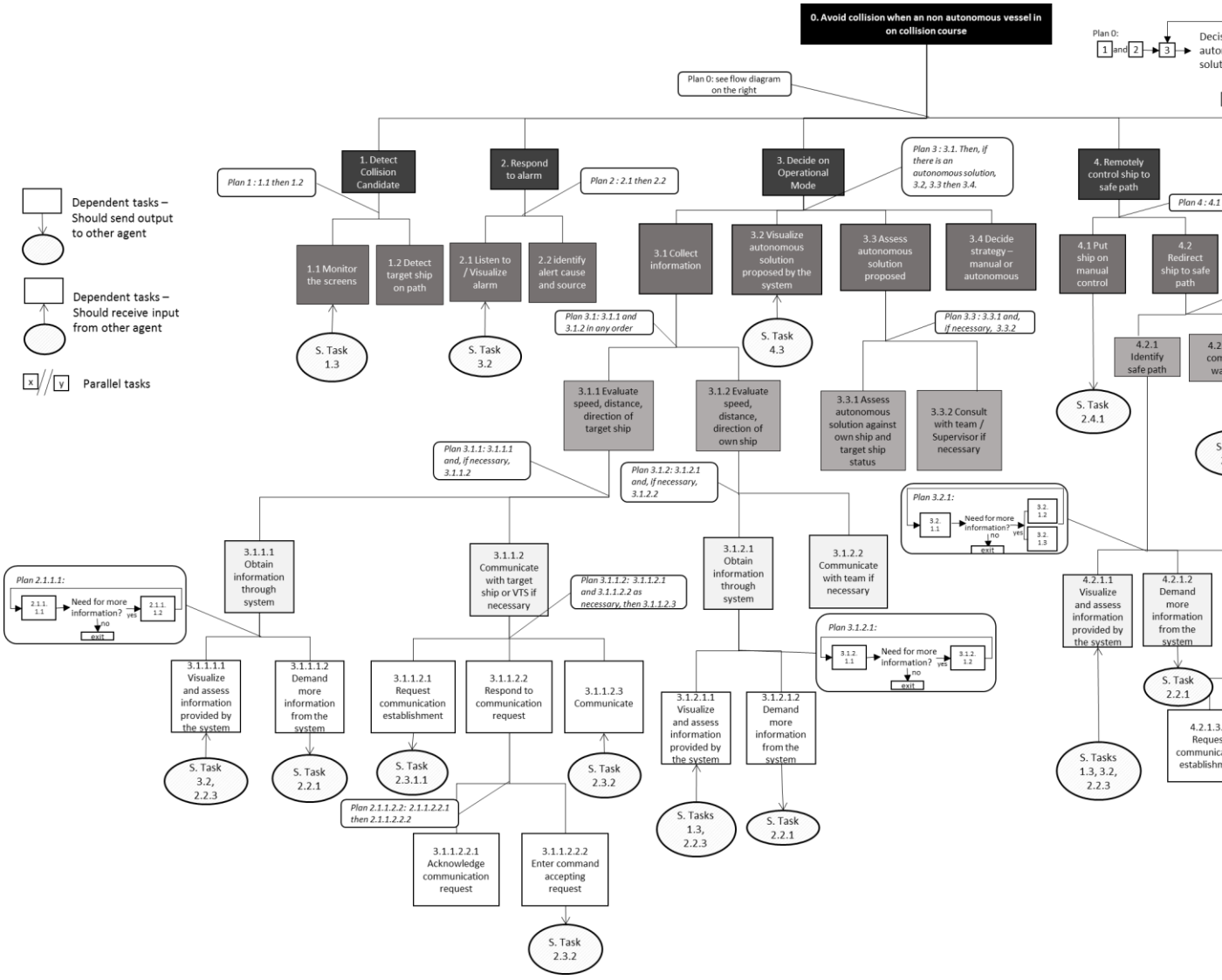


Figure 15 and Figure 16. CoTA could also be used for other cases, such as human-software-hardware interaction.

Table 4: example of identification of propagation of failure and recovery tasks using CoTA

Failure in	**Leads to**	**Failure in …**	**Possible recovery task**
AS task 2.2 (send alarm)	➔	Human task 1.1 (visualize alarm)	Human task 1.1.1 (monitor screens) ← AS task 1.3 (send navigational data)
Human task 3.1 (command change mode)	➔	AS task 6.4.1 (receive command to change operation mode)	No recovery expected – the scenario is ongoing and the recovery would be to repeat Human task 3.1, but there is not enough response time. Will lead to collision if the AS plan is non-existent or inadequate

3.2.1 Developing a CoTA from an ESD

The following steps can be used for developing a CoTA from an ESD, exemplified in **Error! Reference source not found.**. The tasks related to the human operator are marked in dark grey, while the ones related to the AS are coloured in light grey. The figure illustrates the connections between events, tasks and agents (Steps 3 and 4), and a parallel task for the AS (Step 5). The stop rules application (Steps 6 and 7) are further exemplified in the case study in the following Section.

1. Define agents to be analyzed (such as human operators and autonomous ship). Each of the agents will have a HTA;
2. Define Task 0: Task 0 of the HTA is to avoid collision and recover successfully from the initiating event;
3. Define agents acting in the events: Analyze each event of the ESD and define which acting agent is involved: e.g., the autonomous ship, the human operator or both;
4. Define High Level Tasks: Each event of the ESD will translate into a high-level task in each of the respective HTAs. It is advised for the analyst to develop a table for correspondence between the Identifier (ID) of the event from the ESD and the Task ID in the CoTA, to facilitate further analysis;
5. Identify parallel tasks: These will be executed at all times during the scenario, and can support the other tasks or be connected to the interaction between the agents, e.g., communication tasks, listening for commands, etc.;
6. Re-describe tasks until reaching stop-rule: The stop rule is to re-describe the tasks until
   i) they are associated with only one of the I-D-A phases and, for the dependent tasks,
   ii) they represent the interaction with an other agent. The dependent tasks are the ones that receive an input from a task of another agent or send an output to it. It is important that the HTAs are developed concurrently to apply the stop rules.

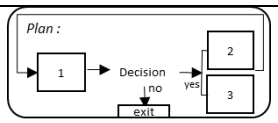
   The analyst should thus use a model based on IDA to re-describe the tasks until reaching only one of the I-D-A phases; and if the low-level task does not represent clearly the agents´ interaction, re-describe it. The input / output of the task must be clearly represented on the HTA of the other agent;

7. Identify interface tasks: Highlight the agents' dependent tasks and their interactions (in Figure 4 the Tasks A.1 and B.1 are dependent)

The CoTA adopts the HTA plans described in [43]. The CoTA plans state how the sub-tasks must be followed to accomplish the main task. The CoTA plan may determine for instance a sequence, or a decision. In addition, it contains the performance of parallel tasks, and a scenario-specific plan. The latter is used in scenario-specific CoTA. Table 5 presents possible plans and their notations. However, the analyst is not restricted to these CoTA plans and notations. The key of developing the CoTA plans

is to include all the information on how the sub-tasks should be performed in order to attain the main goal, in a clear and concise way.

Table 5: Possible task plans and notations for the CoTA

Type	Symbol	Description
Sequence	1→2→3 1 then 2 then 3 1-3	The tasks should be performed in the specific sequence: task 1, then task 2, then task 3.
Sequence	1 and 2 in any order	Tasks 1 and 2 may be performed in any order.
Decision		Task 1 is performed and, according to the decision made, tasks 2 and 3 are performed.
Sequence and Parallel task	[1 → 2] // 3	Task 3 is a parallel task to 1 and 2, which should be performed during the whole operation (at the same time as tasks 1 and 2).
Scenario-specific sequence	[1 → 3] | [2 failed]	This used in a scenario-CoTA (cf. Section 3.2.2); for the sequence of events: success branch of event 1, failure branch of event 2, success branch of event 3. To reach success, then, tasks 1 and 3 should be performed in this order, given that task 2 will fail.

3.2.2 The scenario-specific CoTA

As stated previously, the CoTA may be used for analyzing a specific sequence of events instead of all events of the ESD. For this purpose, the analyst may initially develop the complete CoTA, and then identify the tasks related to the desired sequence of events and re-organize the CoTA. Another option is to develop the CoTA only for that specific sequence of events.

In both cases, the development of the scenario-specific CoTA starts with the identification of the events involved in the desired ESD path. To make use of the complete CoTA and adapt it, the analyst then highlights the tasks of each agent's HTA that belong to that sequence using the Table developed in Step 4 of the CoTA development process(c.f., the example in Section 4). If the analyst does not wish to start with a complete CoTA, then s/he should follow the rules developed in the previous sections only for the desired sequence specific events.

The scenario-specific CoTA merges the HTAs from the agents, including the parallel tasks, and the Plan 0 indicates in which order these tasks should be followed for the sequence of events analyzed. Figure 10 illustrates a scenario-specific CoTA. The sequence of events for analysis if the upper branch of the ESD: events A and B successfully take place, leading to safety. The tasks related to these events are Task A, from the operators' HTA, and Task B, from the AS' HTA. In addition, the AS has a parallel

task, E. The scenario-CoTA has then tasks A, B and E. Plan 0 provides the sequence in which the tasks should be performed: Task A, then Task B; and Task E should be performed all time.

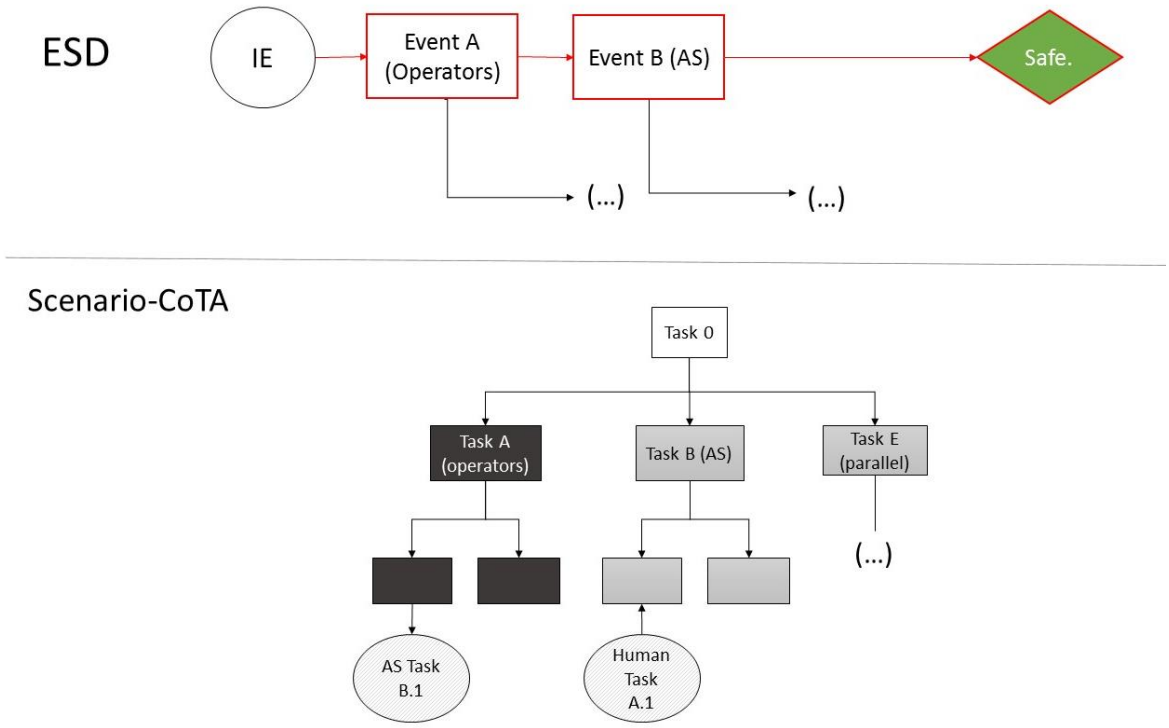


Figure 10: Scenario-specific CoTA example

4 Case Study

4.1 Step 1 – familiarization with the system

The case study consists of the following scenario:

- The autonomous ship is initially operating in constrained autonomy (Level 7 in Table 1): it is responsible for detection and collision avoidance planning and execution. Its operation is supervised by a crew working on a SCC;
    - If the AS fails at detecting the CC, the crew at the SCC can detect it through monitoring the operation;
- The AS detects another ship as a CC and generates a plan. It sends a warning to the SCC about the existence of the CC and the plan generated, in addition to detailed information about the CC;
    - The AS may fail in generating a plan. In this case it can still send a warning to the SCC about the CC;

- The operators evaluate the plan generated by the AS as adequate or inadequate. If it is considered inadequate they can take remote control of the AS;
- If there is no plan generated, the operators' only choice for successful collision avoidance is to take remote control;
- If the CC is a ship and it is behaving in an unexpected manner, the AS and SCC can identify this through the monitoring task, and give commands to correct its actions;
- If the implementation fails, there is not enough time to re-plan and implement a new plan.

These characteristics of the scenario and possible LoA are used to answer the questions for the ESD development, in the next section.

4.2 Step 2: Event Sequence Diagram development

Table 6 presents the questions' answers based on the case study description: Following the answers from Table 6, we build the ESD. The answers determine the existence of the BPs of the ESD - the answer "no" to question 4, for example, determines that the BP "J" does not exist in the ESD. Note that this case study has a dynamic LoA ranging from high (constrained autonomy) to low (remote control). For this reason, the ESD contains most of the BPs from the flowchart, as it has to represent all events from the humans and from the AS. The identification of the branch points of the ESD can be seen in Table 7.

Table 6: Case Study ESD questions

Number	Question	Answer
1	Who is the primary responsible for detection of CC?	AS
2	Can the operators detect the CC from the SCC?	Yes
3	Can the operators remotely control the AS from the SCC?	Yes
4	Can the operator use extreme measures to avoid collision?	No
5	Who is primary responsible for developing collision avoidance plan?	AS
6	Is there an alarm in the SCC warning about the CC, and are they informed on the plan for collision avoidance?	Yes
7	Is the collision candidate a ship?	Yes
8	Is there enough time available to re-plan and implement a new plan to avoid collision?	No

Table 7: Case Study branch points

BP	Name
IE	AS is on collision course
A	Detection of CC by AS
B	Detection of CC by the operators
C	AS generates collision avoidance plans *Outcomes: Adequate plan, no plan, inadequate plan*

D	Operator responds to alarm
E	Decision on operational mode Outcomes: *manual, autonomous*
F	Remote control of the AS by the operators in the SCC
G	Implementation of the collision avoidance plan
H	CC follows collision avoidance plan
I	Monitoring through the AS and SCC, warning the CC; and measures to avoid collision are taken by the CC

AS: autonomous ship; CC: collision candidate; SCC: Shore Control Center

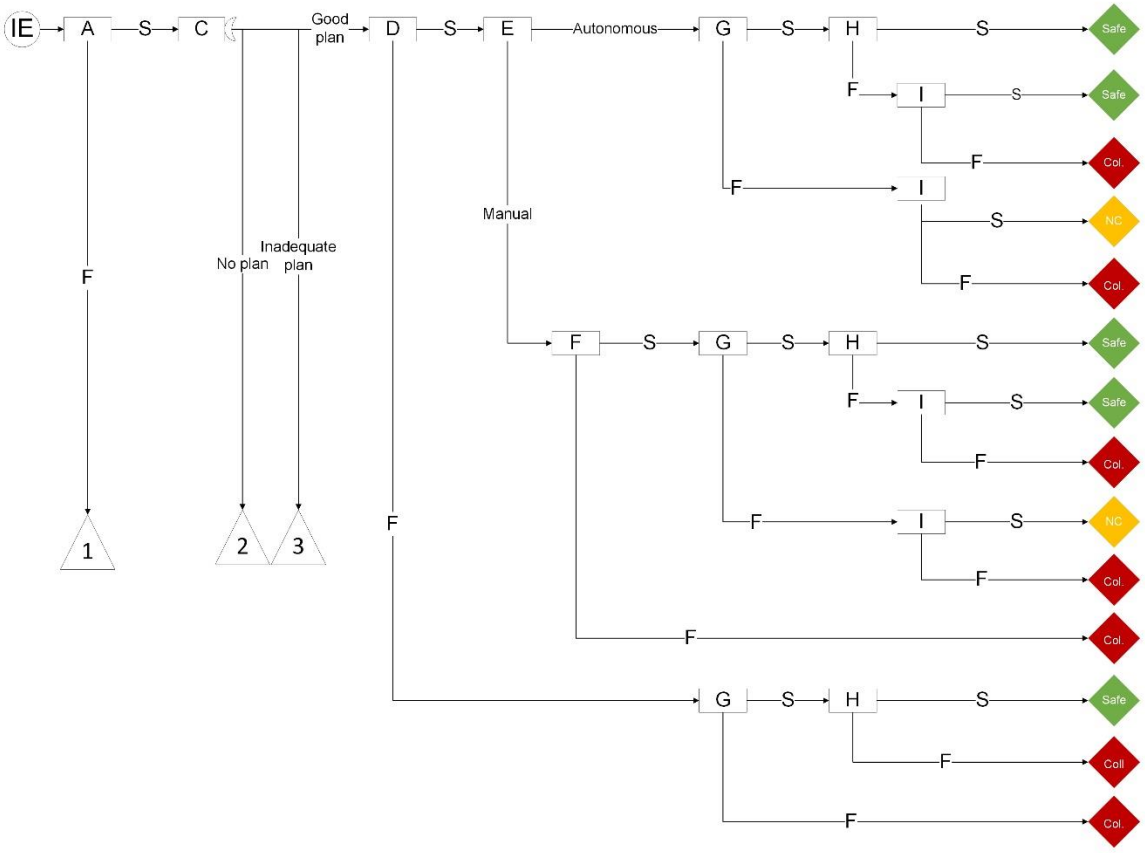


Figure 11(a): Case Study Event Sequence Diagram (cont.)

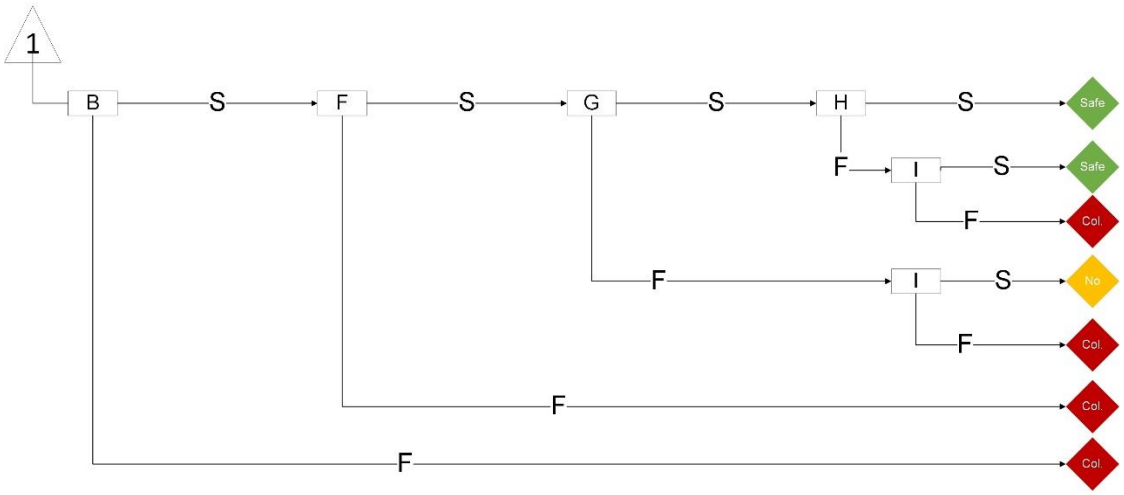


Figure 12(b): Case Study Event Sequence Diagram (cont.)

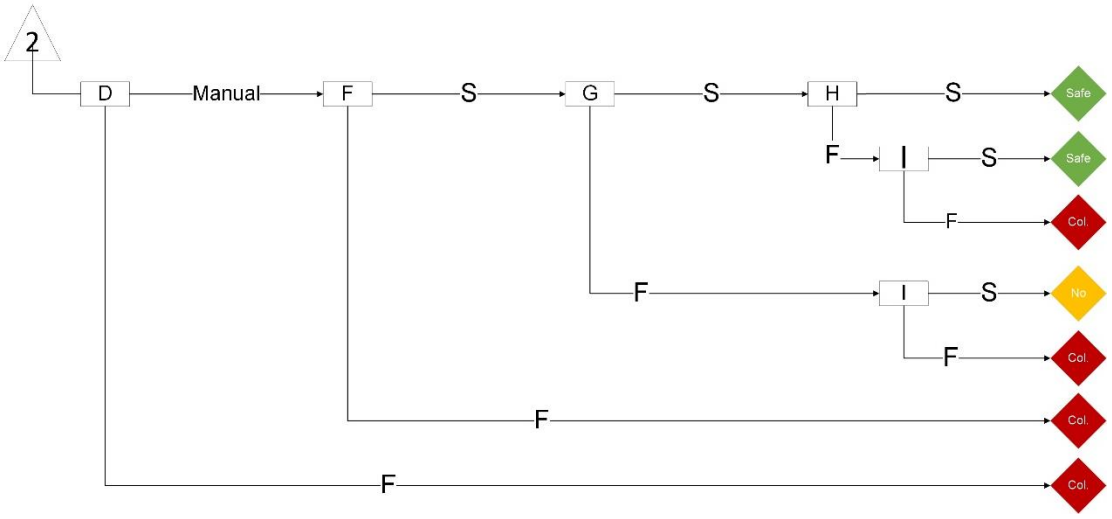


Figure 13(c): Case Study Event Sequence Diagram (cont.)

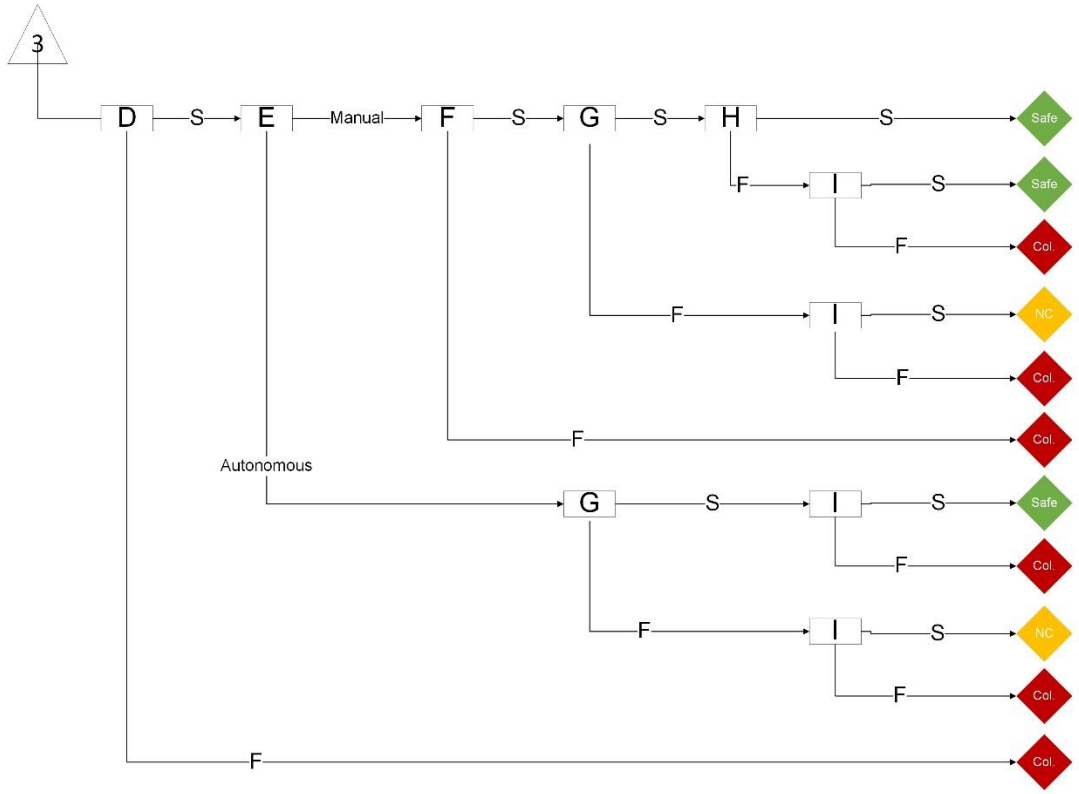


Figure 14(d): Case Study Event Sequence Diagram (final)

4.3 Step 3: Concurrent Task Analysis development

To present an application as detailed as possible, the CoTA developed for the case study is a complete CoTA, i.e., it contains all the events from the ESD. The agents in this case are operators working on a SCC and the AS (Step 1). Task 0 is to avoid collision when a vessel is on collision course (Step 2). The agents acting in each event can be seen in Table 8 (Step 3). Note that the event of monitoring (BP I) can be performed by the SCC and the AS. It will thus be part of both HTAs.

Table 8: Acting agents of the events and ESD-CoTA correspondence

Event	**Acting Agent**	**Task ID**
A	AS	Aut. Ship Task 2
B	SCC	Human Task 1
C	AS	Aut. Ship Task 3
D	SCC	Human Task 2
E	SCC	Human Task 3
F	SCC	Human Task 4
G	AS	Aut. Ship Task 4
H	CC	-
I	SCC / AS	Human Task 5 Aut. Ship Task 5

For step 4, the high-level tasks for each HTA are defined through the correspondence between the BP and the Task ID in the HTAs is indicated in Table 8. The operators' HTA will thus have as high-level tasks to detect the vessel on collision course, respond to the alarm, decide on operational mode, remote control the ship and monitor. The HTA of the AS will have as high-level tasks to detect ship on collision course, generate a collision avoidance plan, implement the plan, and monitor. Steps 5 to 7 are described in the following sub-sections for the operators and for the AS.

4.3.1 HTA Operators

The HTA of the operators has five high level tasks: detect collision candidate, respond to the alarm, decide on operational model, remotely control the ship and monitor.

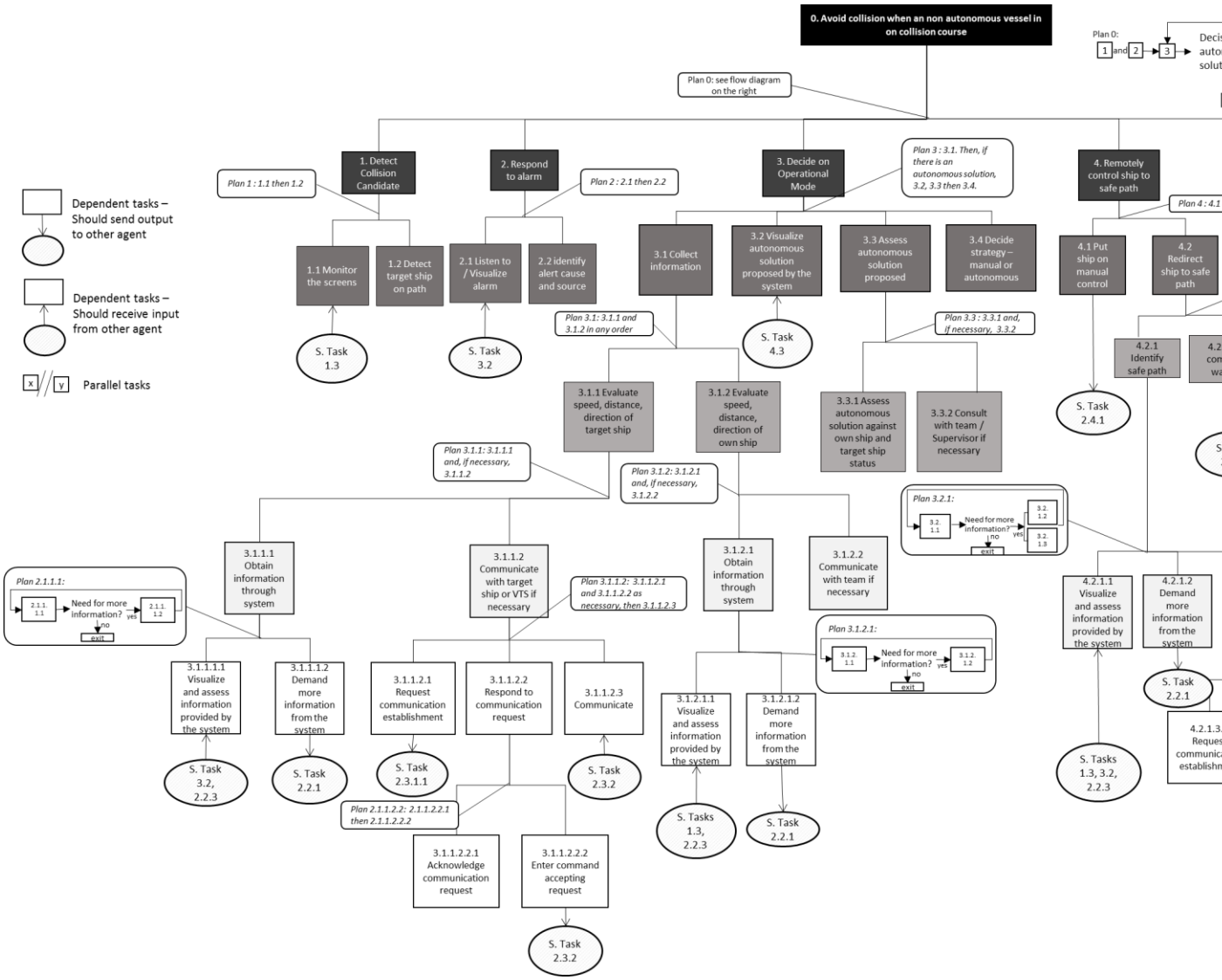


Figure 15 presents the HTA for the operators, adapted from [12].

With respect to step 5, the HTA for the operators does not have a parallel task. However, if monitoring by operators was not an event included in the ESD, the analyst should consider including it as a parallel task in the HTA, since it is a task that must be performed by the operators all time and supports the other tasks.

i) The re-description of the tasks using the stop-rules (Step 6) is as follows: The tasks have been re-described until being associated to only one of the I-D-A phases. Detect collision candidate, for instance, consists of monitoring the screen, which is information gathering - I phase, and detect target ship on path, which is a situation assessment – D phase.

ii) The interface tasks have been identified, and the ones that would not directly represent the interaction between the agents have been re-described. Respond to the alarm, for instance, is within the I-phase – information gathering and pre-processing. However, it comprises of listening/ visualizing the alarm – which depends not only on the cognition and availability of the operator and organizational factors, but also on the AS sending the data to the SCC, which is an interface task; and of identifying the alarm source and cause. It is then re-described to clearly represent the interaction.

The dependent tasks are highlighted with a circle, which also presents the tasks from the other agent (the AS) that have dependency on these (Step 7).

4.3.2 HTA Autonomous Ship

The HTA of the AS has four high level tasks coming from the ESD: detect collision candidate, generate collision avoidance plan, implement the plan and monitoring. In this case study, the AS also has two parallel tasks, identified in Step 5. Two main parallel tasks have been identified for the AS: collect information and listen and answer to commands and requests. These are executed in parallel with the chain of the three tasks mentioned before.

The Steps 6 and 7 follow in a similar manner as described for the operators' HTA in the previous sub-section. The developed CoTA is further discussed in Section 5.

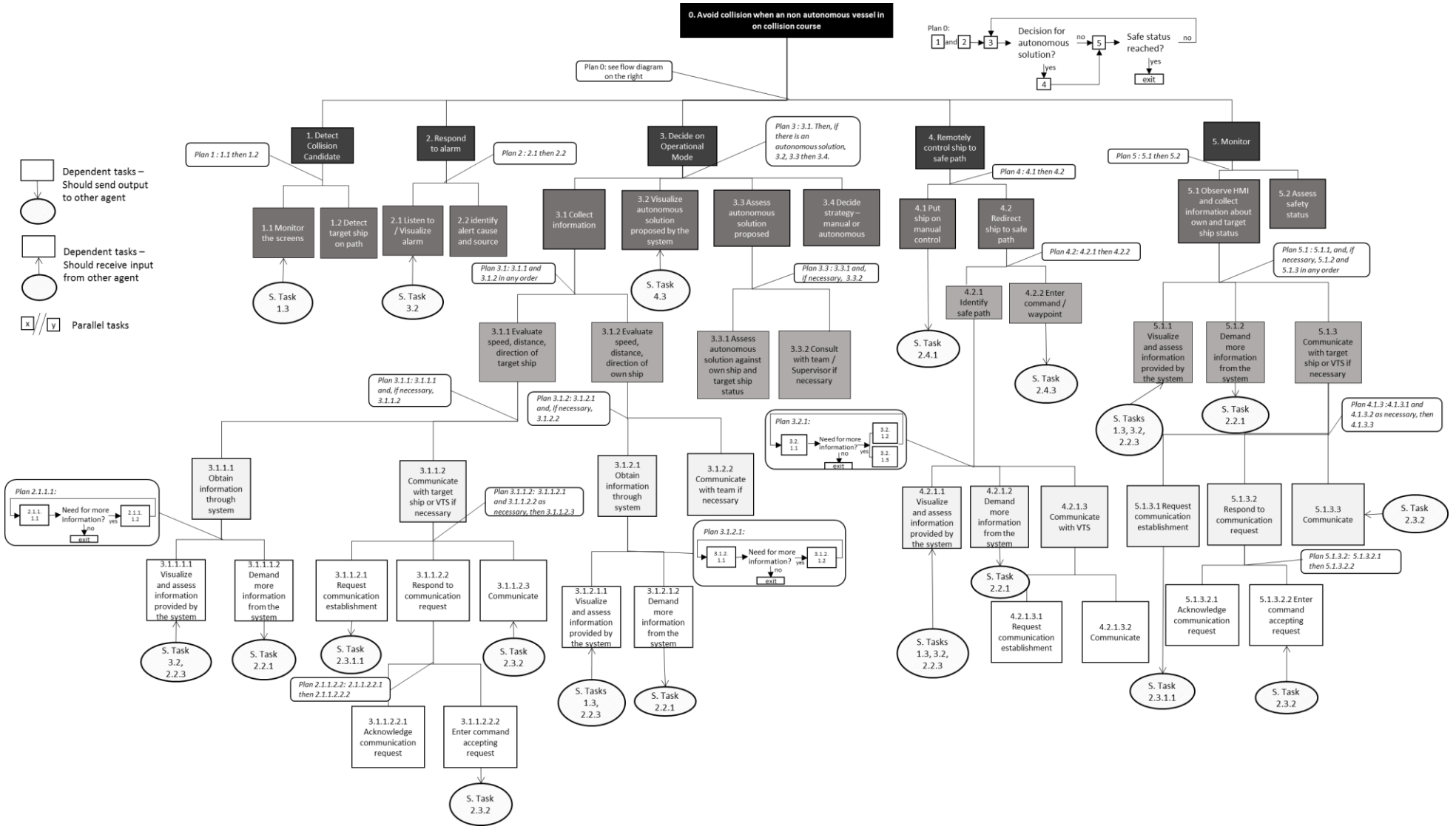


Figure 15: HTA for the operators

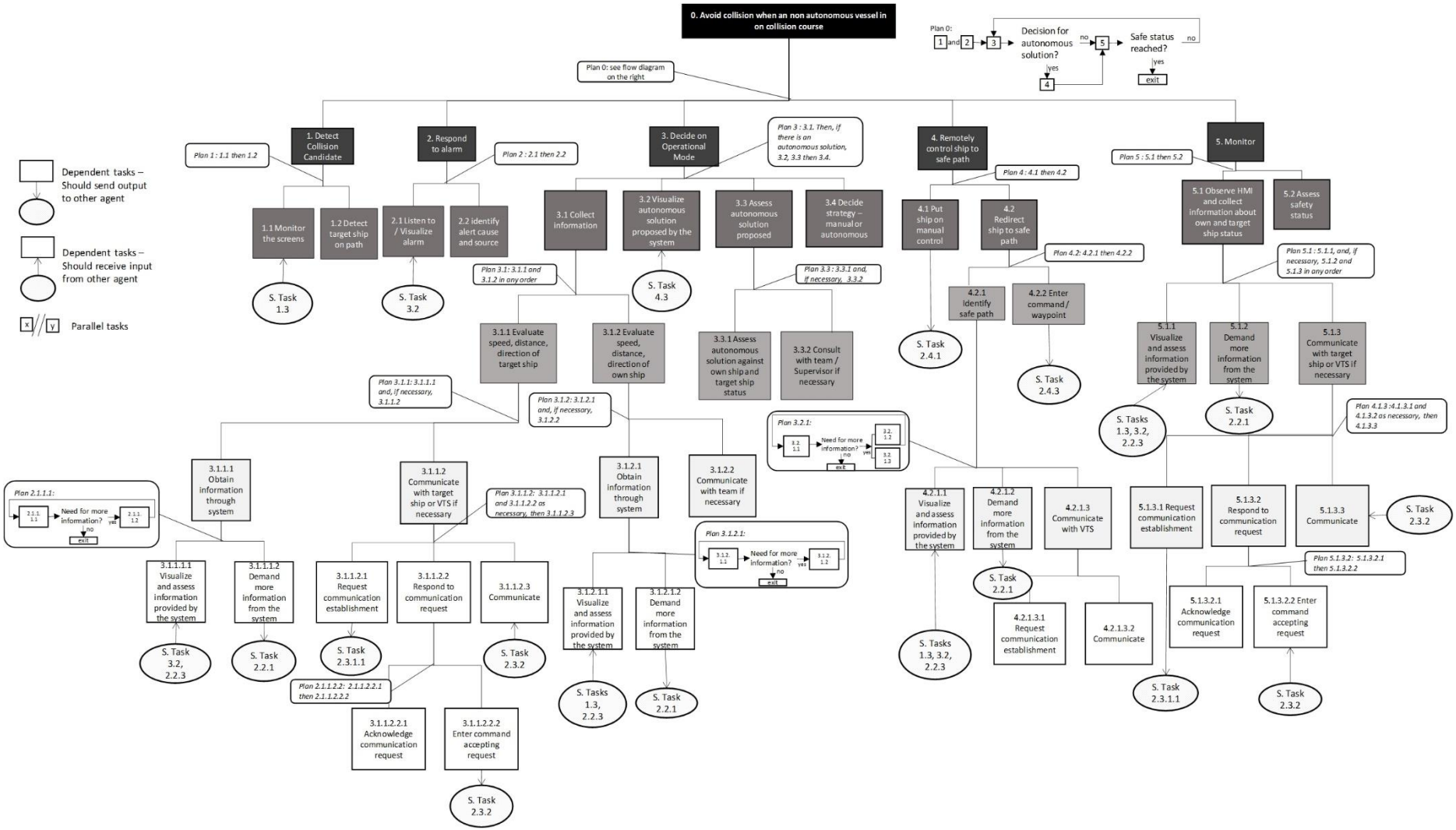


Figure 16: HTA of the Autonomous Ship

5. Method and case-study discussion

The scenario of the case study is relatively complex, since the AS operates with a dynamic LoA that can, according to the situation, shift from low LoA (remote control) to high (constrained autonomy). The application of the flowchart for the ESD development ensures that the foreseeable high level events related to the AS and to the operators are covered. The events from the ESD are successfully translated into tasks and developed into the CoTA using the rules provided in section 3.2.1. This allows for identifying dependent tasks between the agents. A more detailed analysis of the dependent tasks can reveal propagation of failure. This is not in the scope of this work, but an example is provided in Table 4.

The tasks in the CoTA in this analysis were re-described using the stop rules. The tasks are associated with only one IDA phase and clearly represent the dependent tasks. As stated in Section 3.2, the CoTA can be used to identify failure events from the agents involved. Table 10Table 11 in the Appendix present the failure events derived from the CoTA. Table 8 was adapted from [12]. As stated in sub-section 3.2, the Failure events are defined for the lower level tasks, and include the alternatives "not performing the task", "performing it inadequately", and "performing it too early/ late". It comprises thus errors of omission and commission.

Depending on the goal of the analysis, the tasks in the CoTA could be further decomposed. For instance, if the goal is to identify the sole responsible component for a task of the AS, the task of manoeuvring could be re-described as determining necessary control input in the control system, sending control input to the machinery, and executing the manoeuvre through changes in rudder angle and propeller speed . Software risk assessment models could be used to analyse the software in more detail, revealing in-depth software failure modes and failure propagation, c.f. [44], [45].

The case study's CoTA in this paper contains all events of the ESD, i.e., it presents all tasks that must be performed by each agent to be successful. If the analyst needs to have the tasks associated with only one specific path of the ESD, s/he can develop a scenario-specific CoTA, using the full CoTA as a starting point.

An interesting scenario for analysis is when the autonomous ship is successful in its tasks and this will lead to a safe state, even if the operator is not aware of the potential collision and does not respond to the alarm. The scenario-specific CoTA will present then tasks that must be performed successfully by the AS in order to avoid collision independently of human error in acknowledging the potential collision. The sequence of events in this scenario is following the successful branches of A and C, the failure branch of D, followed by the successful branches of G and H (c.f. **Error! Reference source not found.**). The corresponding tasks are presented in Table 9. The CoTA will thus be composed of these

events and, additionally, of the parallel tasks of the AS. As illustrated in Figure 10, the CoTA merges the Human's and AS' HTAs. The CoTA Plan 0, in this case, will be: the system shall perform tasks 2, 3 and 4 in this order, given that human task 2 failed; and shall perform tasks 1 and 6 all time. Using the notation from Table 5, it can be written as: "Plan 0: [S.T. 2 → S.T. 3 → S.T. 4 | H.T. 2 failed] // S.T. 1 // S.T. 6".

Table 9: Events and Task correspondence for the scenario-specific CoTA

Event	**Task ID**	**Branch**
A	Aut. Ship Task 2	Success
C	Aut. Ship Task 3	Success
D	Human Task 2	Failure
G	Aut. Ship Task 4	Success
H	-	Success

The task failure events from Table 8 and Table 9 of the Appendix can then be used to identify failures that may be involved in this scenario-CoTA, i.e., which failures may prevent reaching success end state in this scenario.

Moreover, the task failure events can be analysed to identify its possible causes (named performance influencing factors in HRA), which can be further modelled using BBNs. In these cases, the interface task could be modelled as a possible causes for a failure event. For example, visualizing the screens (human task 1.1), is dependent on the task from the AS of sending the appropriate data to the SCC (AS

task 1.3), c.f.,

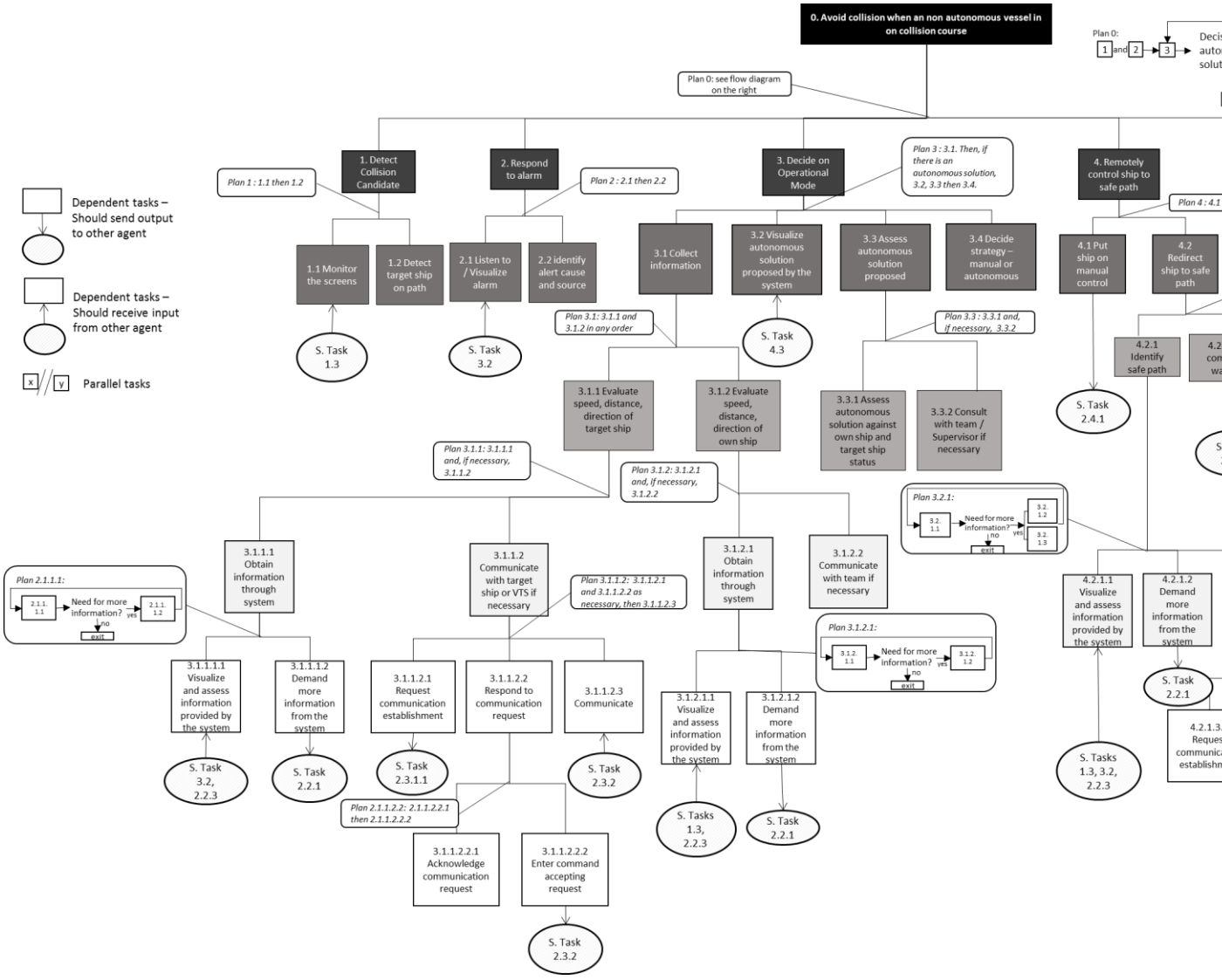


Figure 15 and Figure 16. A failure on checking the screen by the operator to visualize the necessary information could have one or several of the following reasons: inadequate HMI output, extra workload (e.g., operator is monitoring too many simultaneously), low morale/ attitude, low attention, and information not received (AS did not send data to SCC). Indeed, the probability of the operator failing in the task of visualizing information at the screen is 1 if the AS does not send the data to the SCC.

The application of the method to this case study illustrates the strengths of the method. The flowchart is a practical and easy tool to build the ESD, providing traceability to the analysis and ensuring that important events are covered. It can be used to analyse not only ASs designed to operate with a very low or very high LoA, but also ASs that can change their LoA during operation. The events related to all involved agents of the system in the course of a collision avoidance can be included. The ESD flowchart was tested for different LoAs (low to high) and also tested by different researchers for the same scenarios to ensure reproducibility.

The above discussion on the case study CoTA also illustrates its potential for analysis of H-AS collaboration for a diversity of purposes. It identifies important interactions between these agents in a task level, as well as propagation of failures and dependency of the tasks, which may be otherwise overlooked.

Although the Task Failure Events cover errors of commission and omission, they may not map all possible failure events from the AS and the operators. Since they are derived from the CoTA, which is success-oriented, errors such as performing an unexpected task are not identified. However, the CoTA can be paired with other tools to identify these errors.

The features of the ESD and CoTA makes the H-SIA method a valuable technique for analysis of safety of AS operations. It may be used in the design phase, to develop procedures and to derive specifications, for failure events identification, and the results can be further integrated into risk assessments.

Some limitations of the methods are derived from the considerations assumed for building the ESD flowchart, as the conservative approach that the CC cannot take actions to avoid collision unless requested by the AS/SCC. Although the CoTA is developed using clear guidelines and stop-rules, the identification of parallel tasks and the re-description depends also on the analyst. This may lead to different CoTAs when applied by different analysts. This variability is, in one sense, a limitation of the method. On the other hand, it offers flexibility for the CoTA to be developed and detailed according to the purpose of the analysis.

6. Conclusion

This paper presents the Human-System Interaction in Autonomy method for MASS, developed for collision scenarios. The method consists of an ESD and a Concurrent Task Analysis (CoTA) – a multipurpose technique introduced in this method. H-SIA considers the human-autonomous ship collaboration, an important part of autonomous ships operation, as well as the dynamic LoA these operations may have. The H-SIA method allows for analysing the autonomous system as a whole during different phases of design or operation to ensure safe MASSs. It can model different LoAs, and can thus be used to compare safety relevant aspects of autonomous ship navigation phases with different LoAs or between autonomous ships designed with different LoAs.

The use of the flowchart to develop the ESD ensures that the analyst considers all important events arising from the human operators and the AS, and provides a traceable and reproducible analysis. Moreover, the H-SIA method is flexible in the sense that the ESD can be coupled with not only the CoTA, but also to FTs and BBNs, as needed in the analysis. The use of the CoTA, in turn, allows for identifying the tasks the agents must accomplish to successfully perform the events of the ESD. The

CoTA can be used to identify dependent tasks between agents of the system; identify system and human failures; analyse propagation of failure between the agents; and develop procedures and specifications.

Currently the H-SIA method applies to collision scenarios for AS, but it can be extended to other hazard scenarios, such as grounding. H-SIA has been applied to scenarios with one collision candidate. Yet, most of the tasks are similar for more complex scenarios with several collision candidates. Further investigation on the direct applicability is necessary. Future work includes also the development of fault trees and BBNs, in a hybrid causal logic model. The method can benefit from validation through applications to existing AS projects, as well as through feedback from risk and shipping experts use.

Further, an important feature of the H-SIA method is that its principles can be applied for other types of autonomous systems. Autonomous systems in general may have dynamic LoA and can be (remotely) operated by a human Hence, the ESD questions, the ESD flowchart and the CoTA could be adapted to other autonomous systems. This is subject to future research work.

7. References

[1] Kongsberg, "Autonomous ship project, key facts about YARA Birkeland," 2017. [Online]. Available: https://www.km.kongsberg.com/ks/web/nokbg0240.nsf/AllWeb/4B8113B707A50A4FC125811D00407045?OpenDocument.

[2] H. Alfheim and K. Muggerud, "Development of a Dynamic Positioning System for the ReVolt Model Ship Henrik Alfheim," NTNU, 2017.

[3] NTNU, "Autoferry – Autonomous all-electric passenger ferries for urban water transport," 2018. [Online]. Available: https://www.ntnu.edu/autoferry. [Accessed: 13-Aug-2018].

[4] M. Laurinen, "Remote and Autonomous Ships: The next steps," *AAWA Adv. Auton. Waterborne Appl.*, p. 88, 2016.

[5] Ø. Rødseth and H. Nordahl, "Definitions for Autonomous Merchant Ships," 2017.

[6] Ø. J. Rødseth, "Definition of autonomy levels for merchant ships," 2017.

[7] MUNIN, "Research in maritime autonomous systems project results and technology potentials," 2016.

[8] AGCS, "Safety and Shipping Review 2017," 2017.

[9] A. M. Rothblum, "Human Error and Marine Safety," in *National Safety Council Congress and Expo*, 2000.

[10] Ø. J. Rødseth and A. Tjora, "A risk based approach to the design of unmanned ship control

systems," *Proceeding Conf. Marit. Technol.*, pp. 153–162, 2014.

[11] M. Ramos, I. B. Utne, J. E. Vinnem, and A. Mosleh, "Accounting for human failure in autonomous ships operations," 2018, pp. 355–363.

[12] M. Ramos, I. B. Utne, and A. Mosleh, "Collision avoidance on maritime autonomous surface ships: operators' tasks and human failure events," *Saf. Sci.*, vol. 116, pp.33-44, 2018.

[13] Ø. J. Rødseth and Å. Tjora, "A risk based approach to the design of unmanned ship control systems," in *Maritime-Port Technology and Development*, 2017, pp. 153–162.

[14] C. A. Thieme, I. B. Utne, and S. Haugen, "Assessing ship risk model applicability to Marine Autonomous Surface Ships," *Ocean Eng.*, vol. 165, no. February, pp. 140–154, 2018.

[15] P. T. Pedersen, "Review and application of ship collision and grounding analysis procedures," *Mar. Struct.*, vol. 23, no. 3, pp. 241–262, 2010.

[16] Maritime Unmanned Navigation through Intelligence in Networks, "The Shore Control Centre Navigational shore support : A new perspective The Shore Control Centre."

[17] S. Li, Q. Meng, and X. Qu, "An Overview of Maritime Waterway Quantitative Risk Assessment Models," *Risk Anal.*, vol. 32, no. 3, pp. 496–512, 2012.

[18] G. J. Lim, J. Cho, S. Bora, T. Biobaku, and H. Parsaei, "Models and computational algorithms for maritime risk analysis: a review," *Ann. Oper. Res.*, pp. 1–22, 2018.

[19] Ø. J. Rødseth and H.-C. Burmeister, "Risk Assessment for an Unmanned Merchant Ship," *TransNav, Int. J. Mar. Navig. Saf. Sea Transp.*, vol. 9, no. 3, pp. 357–364, 2015.

[20] J. Kretschmann, L., Rødseth, Ø. J., Tjora, Å., Sage Fuller, B., Noble, H. & Horahan, "D9.2: Qualitative Assessment. Maritime Unmanned Navigation through Intelligence in Networks," 2015.

[21] Maritime Unmanned Navigation through Intelligence in Networks, "D9.3 : Quantitative assessment," 2015.

[22] F. Jensen, "Hazard and Risk Assessment of Unmanned Dry Bulk Carriers on the High Seas," Technische Universität Hamburg, 2015.

[23] K. Wrobel, P. Krata, J. Montewka, and T. Hinz, "Towards the Development of a Risk Model for Unmanned Vessels Design and Operations," *TransNav, Int. J. Mar. Navig. Saf. Sea Transp.*, vol. 10, no. 2, pp. 267–274, 2016.

[24] K. Wr??bel, J. Montewka, and P. Kujala, "Towards the assessment of potential impact of unmanned vessels on maritime transportation safety," *Reliab. Eng. Syst. Saf.*, vol. 165, no.

August 2016, pp. 155–169, 2017.

[25] B. Rokseth, I. B. Utne, and J. E. Vinnem, “A systems approach to risk analysis of maritime operations,” *Proc. Inst. Mech. Eng. Part O J. Risk Reliab.*, vol. 231, no. 1, pp. 53–68, 2017.

[26] B. Rokseth, I. B. Utne, and J. E. Vinnem, “Deriving verification objectives and scenarios for maritime systems using the systems-theoretic process analysis,” *Reliab. Eng. Syst. Saf.*, vol. 169, no. March 2017, pp. 18–31, 2018.

[27] O. A. V. Banda and S. Kannos, “Hazard Analysis Process for Autonomous Vessels,” 2017.

[28] I. Utne, I. B., Sørensen, A. J. & Schjølberg, “OMAE2017-61645,” in *Proceedings of the ASME 2017 36th International Conference on Ocean, Offshore and Arctic Engineering, OMAE 2017*, 2017, pp. 1–10.

[29] S. Swaminathan and C. Smidts, “The Event Sequence Diagram framework for dynamic Probabilistic Risk Assessment,” *Reliab. Eng. &andSystems Saf.*, vol. 63, pp. 73–90, 1999.

[30] N. J. Ekanem, A. Mosleh, and S.-H. Shen, “Phoenix–A model-based Human reliability analysis methodology: Qualitative analysis procedure,” *Reliab. Eng. Syst. Saf.*, vol. 145, pp. 1–15, 2015.

[31] N. J. Ekanem and A. Mosleh, “Phoenix – A Model-Based Human Reliability Analysis Methodology : Quantitative Analysis Procedure and Data Base,” in *Proceedings to the Probabilistic Safety Assessment and Management PSAM 12*, 2014.

[32] M. A. Ramos, “A Methodology for Human Reliability Analysis of Oil Refineries and Petrochemical Plants,” Federal University of Pernambuco, 2017.

[33] W. Røed, A. Mosleh, J. Erik, and T. Aven, “On the use of the hybrid causal logic method in offshore risk analysis,” vol. 94, pp. 445–455, 2009.

[34] K. Groth, C. Wang, and A. Mosleh, “Hybrid causal methodology and software platform for probabilistic risk assessment and safety monitoring of socio-technical systems,” *Reliab. Eng. Syst. Saf.*, vol. 95, no. 12, pp. 1276–1285, 2010.

[35] S. Swaminathan and C. Smidts, “Mathematical formulation for the event sequence diagram framework,” *Reliab. Eng. Syst. Saf.*, vol. 65, no. 2, pp. 103–118, 1999.

[36] A. Shepherd, *Hierarchical Task Analysis*. London: Taylor & Francis, 2001.

[37] J. Annett and N. Stanton, *Tasks Analysis*. 2000.

[38] D. Diaper and N. Stanton, *The Handbook of Task Analysis for Human-Computer Interaction*. 1992.

[39] B. Kirwan, *A guide to practical human reliability assessment*. London: Taylor & Francis, 1994.

[40] B. Stroustrup, *A Tour of C++*. Upper Saddle River, NJ, USA: Pearson Education inc., 2014.

[41] C. Smidts, S. H. Shen, and A. Mosleh, "The IDA cognitive model for the analysis of nuclear power plant operator response under accident conditions. Part I: problem solving and decision making model," *Reliab. Eng. Syst. Saf.*, vol. 55, no. 1, pp. 51–71, 1997.

[42] T. Statheros, G. Howells, and K. McDonald-Maier, "Autonomous ship collision avoidance navigation concepts, technologies and techniques," *J. Navig.*, vol. 61, no. 1, pp. 129–142, 2008.

[43] J. Annett, "Hierarchical Task analysis," in *The Handbook of Task Analysis for Human-Computer Interaction*, D. Diaper and N. A. Stanton, Eds. London: Lawrence Erlbaum Associates, 2008, pp. 67–82.

[44] C. A. Thieme, A. Mosleh, I. B. Utne, and J. Hegde, "Incorporating Software Failure in Risk Analysis – Part 1: Software Functional Failure Mode Classification.," *Submitt. to Reliab. Eng. Syst. Saf.*

[45] C. A. Thieme, A. Mosleh, I. B. Utne, and J. Hegde, "Incorporating Software Failure in Risk Analysis – Part 2: Risk Modeling Process and Case Study," *Submitt. to Reliab. Eng. Syst. Saf.*

APPENDIX A – Task failure events deriving from the case study CoTA

Table 10: Human Task Failure Events derived from CoTA (adapted from *[12]*)

Task ID	Task	Human Task Failure Events
1.1	Monitor the screens	Information on the screens not visualized
1.2	Detect collision candidate on path	Collision candidate not visualized
		Collision candidate not recognized as a collision candidate
		Collision candidate recognized too late
2.1	Listen to / visualize alarm	Alarm not listened/ visualized
		Alarm listened / visualized too late
2.2	Identify alarm cause and source	Alarm cause / source not identified
		Alarm associated to different cause / source
		Alarm cause / source not identified on time
3.1.1.1.1, 3.1.2.1.1, 4.2.1.1	Visualize information provided by the system	Information provided by the system not visualized
		Visualized information from the wrong source / ship
		Information provided by the system misread
		Information provided by the system misunderstood
3.1.1.1.2, 3.1.2.1.2, 4.2.1.2	Demand more information to the system	Failure in acknowledging the need for more information
		Wrong command sent to the system
		Command sent to the system wrongly
		Command not sent to the system
		Information provided by the system misread
		Information provided by the system misunderstood
3.1.1.2.1, 4.2.1.3.1, 5.1.3.1	Request Communication establishment	Failure in acknowledging the need for communication
		Wrong command sent to the system
		Command sent to the system wrongly
		Command not sent to the system
3.1.1.2.2.1, 5.1.3.2.1	Acknowledge communication request	Request not visualized
		Request not responded to
		Request not responded to on time
3.1.1.2.2.2, 5.1.3.2.2	Enter command accepting request	Wrong command sent to the system
		Command sent to the system wrongly
		Command not sent to the system
		Command sent to the system too late
3.1.1.2.3, 4.2.1.3.2, 5.1.3.3	Communicate	Information miscommunicated
3.1.2.2.	Communicate with team if necessary	Failure in acknowledging the need for communication with the team
		Information miscommunicated

Task ID	Task	Human Task Failure Events
3.2	Visualize autonomous solution proposed by the system	Autonomous solution not visualized
		Autonomous solution misunderstood
3.3.1	Assess autonomous solution against own ship and collision candidate status	Situation misdiagnosed
3.3.2	Consult with team / Supervisor if necessary	Failure in acknowledging the need for communication with the team
		Information miscommunicated
3.4	Decide strategy – manual or autonomous	Inappropriate strategy chosen
4.1	Put ship on manual control	Action on wrong ship
		Command not sent to the system
		Wrong command sent to the system
		Command sent to the system too late
5.2	Assess safety status	Situation misdiagnosed

Table 11: Autonomous Ship Task Failure events derived from CoTA

Task ID	Task	Autonomous Ship Task Failure Events
1.1	Collect raw data	No data collected
		Data collected imprecisely
		Data collected wrongly
		Data collected too late
1.2.1	Process data and estimate ship condition	No ship condition estimated
		Ship condition estimated inaccurately
		Ship condition estimated wrongly
1.2.2	Process data and assess the traffic situation	Traffic situation is not assessed
		Traffic situation is assessed incorrectly (no existing obstacles detected)
		Traffic situation is assessed incorrectly (existing obstacles not detected)
		Traffic situation assessed too late
1.2.2.1	Identify and classify vessels and objects	Vessels and objects are not classified
		Vessels and objects are identified wrongly
		Vessels and objects are classified wrongly
		Vessels and objects and classified too late
1.2.2.2	Determine position and direction of movement of vessels and objects	No position or directions are identified
		Positions are determined wrongly
		Directions are determined wrongly
		Positions are determined imprecisely
		Directions are determined imprecisely
		Positions and directions determined too late
1.2.3	Estimate trajectories of all vessels	Trajectories are not estimated
		Trajectories are estimated incorrectly
		Trajectories are estimated imprecisely

Task ID	Task	Autonomous Ship Task Failure Events
1.3	Send position and navigational data to SCC	Data not send to SCC
		Incorrect data send to SCC
		Position and navigational data send too late
2.1	Establish and maintain communication with SSC	Communication not established
		Communication line established is insufficient (data transfer rate, delays)
		Communication is interrupted (not maintained)
2.2.1	Receive request for data	Requests not received
		Requests received wrongly
		Requests received too late
2.2.2	Retrieve and prepare requested data	Data cannot be retrieved
		Wrong data retrieved and prepared
		Data retrieved too late
2.2.3	Transfer data	Data not transferred
		Data transferred wrongly
		Incorrect data transferred
		Data transferred too late
2.3.1.1	Receive command from SSC	Command not received
		Command not received correctly
		Command received too late
2.3.1.2	Receive command from ship on collision course	Command not received
		Command not received correctly
		Command received too late
2.3.1.3	Receive command from VTS	Command not received
		Command not received correctly
		Command received too late
2.3.2	Establish communication line between collision candidate/ VTS and SCC	Communication line cannot be established
		Communication line established wrongly (incorrect participants)
		Communication line established too late
2.4.1	Receive command to change operational mode	Command not received
		Command not received correctly
		Command received too late
2.4.2	Change operational mode	Operational mode not changed
		Operational mode changed to wrong mode
2.4.3	Obtain command from SCC operator	Command not received
		Command not received correctly
		Command received too late

Task ID	Task	Autonomous Ship Task Failure Events
2.4.4	Update mission plan with the operator input	Mission plan not updated
		Mission plan updated incorrectly
		Mission plan updated too late
3.1	Detect collision candidates	No collision candidates detected
		Object or vessel wrongly identified as collision candidate
		Collision candidate detected that are not on collision course while overlooking other collision candidates
		Collision candidates detected too late
3.2	Inform the operators in the SCC on situation and relevant information on collision candidate	No data sent
		Wrong data sent
		Data send too late
4.1	Decide on applicable COLREG and local rules	No applicable rules decided on
		Wrong decision on applicable rules
		Using standard rules ignoring local rules
4.2	Determine optimal avoidance route, according to applicable rule	No avoidance route was found
		Avoidance route is not adequate
		Avoidance route is not optimal
		Route determined too late
4.3	Inform the SCC on the planned strategy	Strategy is not sent to the SCC
		Strategy is sent incorrectly to the SSC
5.1	Determine necessary control input to the machinery	Control input is not determined
		Control input is determined incorrectly (e.g. random control input, control input leading to under actuation, control input leading to over actuation)
		Control input determined too late
5.2	Send the control input to the machinery and execute maneuver	Control input not sent
		Control input sent wrongly (too late, too early, wrong format, wrong recipient)
		Maneuver not executed
		Maneuver not executed correctly
		Maneuver executed too late
6.1	Monitor collision candidate	Collision candidate movements not monitored
		Collision candidate movements monitored incorrectly
6.2	Assess if safety is established	Safety not assessed
		Safety incorrectly assessed as established