ISBN 978-82-326-3481-1 (printed version)
ISBN 978-82-326-3480-4 (electronic version)
ISSN 1503-8181

NTNU
Norwegian University of
Science and Technology

Christoph Alexander Thieme

Doctoral Thesis

Doctoral theses at NTNU, 2018:346

NTNU
Norwegian University of
Science and Technology
Faculty of Engineering
Department of MarineTechnology

NTNU

Doctoral theses at NTNU, 2018:346

Christoph Alexander Thieme

Risk Analysis and Modelling of Autonomous Marine Systems

NTNU
Norwegian University of
Science and Technology

Christoph Alexander Thieme

Risk Analysis and Modelling of Autonomous Marine Systems

Thesis for the degree of Philosophiae Doctor

Trondheim, October 2018

Norwegian University of Science and Technology
Faculty of Engineering
Department of MarineTechnology

NTNU
Norwegian University of
Science and Technology

NTNU
Norwegian University of Science and Technology

Thesis for the degree of Philosophiae Doctor

Faculty of Engineering
Department of MarineTechnology

ISBN 978-82-326-3481-1 (printed version)
ISBN 978-82-326-3480-4 (electronic version)
ISSN 1503-8181

Doctoral theses at NTNU, 2018:346

Printed by Skipnes Kommunikasjon as

‘Tell me and I forget.
Teach me and I remember.
Involve me and I learn’.

- Benjamin Franklin

This page is intentionally left blank

Preface

This thesis is submitted in partial fulfilment of the requirements for the degree of Philosophiae Doctor (PhD) at the Norwegian University of Science and Technology (NTNU) in Trondheim, Norway. The work was carried out in association with the NTNU Centre of Excellence (SFF, Norwegian: Senter for fremragende forskning) for Autonomous Marine Operations and Systems (AMOS), which is funded through the Research Council of Norway through the SFF funding scheme, Project number 223254 – NTNU AMOS.

The PhD work has been carried out at the Department of Marine Technology (IMT) at NTNU. The main supervisor was Professor Ingrid B. Utne. Professor Ingrid Schjølberg was co-supervising from the beginning of the PhD period. Professor Ali Mosleh from the B. John Garrick Institute of the Risk Sciences at the University of California in Los Angeles (UCLA), Henry Samueli School of Engineering and Applied Science was appointed as co-supervisor in February 2018. During spring 2017, research was carried out with Professor Mosleh at the B. John Garrick Institute of the Risk Sciences at UCLA.

This thesis targets readers across several fields, and the foremost are designers, risk analysts, and operators of autonomous systems being used in a marine setting. This is not an exclusive audience since the principles and findings in this thesis may apply to other autonomous systems, highly automated systems, or parts of these. The presented results may influence the future perspective that is taken during design and operation of such systems.

When I started my PhD research in August 2014, I had completed my master studies at the IMT at NTNU. In my master thesis, I focused on risk and reliability assessment of remotely operated vehicles and autonomous underwater vehicles. During the course of my PhD work, it became apparent that the marine and maritime industry is undergoing a significant change by developing autonomous vessels and ships. This trend is reflected in this dissertation since the first studies focused on underwater vehicles. During the second half of the research period, the general challenges associated with the operation of autonomous ships and vessels were addressed.

Trondheim, October 2018

This page is intentionally left blank

Summary

Autonomous marine systems (AMSs) are of increasing interest for the marine and maritime industries. AMSs are engineered, computer-controlled systems that take (to some degree) decisions independent of their human operators. Different types of AMSs can be differentiated, for example, maritime autonomous surface ships (MASSs), autonomous underwater vehicles (AUVs), or unmanned surface vehicles (USVs). AMSs reduce the operational cost, the risk with respect to personnel, and the energy consumption in comparison to their conventional equivalents. AUVs are already in use and MASSs are expected to be in operation before 2020 (Kongsberg Maritime, 2017). To accept these systems, the public and authorities require that they are safe and do not have higher levels of risk than conventional systems (Danish Maritime Authority, 2018).

The objective of this thesis is to present risk analysis and risk modelling approaches for AMSs. These risk models and risk modelling approaches assist in demonstrating that AMSs are as safe as required and provide decision support during the design and operation of AMSs. This thesis addresses three issues: (i) Identification of risk-influencing factors for AMSs, (ii) presentation of risk analysis and risk modelling approaches for AMSs, and (iii) description of a risk monitoring approach for the operation of AMSs.

Risk assessments are used to analyse and evaluate the level of risk through risk models and suggest improvement measures to reduce the level of risk if necessary (Rausand, 2011). In this thesis, current risk models and approaches have been reviewed to evaluate their applicability for AMSs. AMSs have recently received more attention with respect to their development and design. Only a few risk modelling approaches exist for AMSs. It was found that software and the human operators are not considered in sufficient detail in current risk models for AMSs.

A process to incorporate the risk contribution from software into risk analysis is presented in this thesis. The process relies on the functional decomposition of software, identification of failure modes for the functions, and assessment of the effect of the failure modes on the software output through failure mode propagation. The functional level of software is defined. In addition, a functional failure mode taxonomy for software is developed from the literature. This is necessary since the current taxonomies are not coherent with respect to their level of system application, for example, the overall system level or functional level.

The identified effects on the software output are related to the effect on the external interfaces of the software, for example, human operators, other software systems, or actuators. These effects can be included in risk models, such as fault trees, event trees, or Bayesian belief networks (BBN).

This thesis also addresses the interaction between the human operators and the AMSs in risk analysis. First, the necessity to consider these interactions is highlighted in a risk management framework for AUVs. The framework identifies two phases of risk management where the human operators need to be considered; this is during risk analysis and during the identification of risk-mitigating measures.

Second, a risk model using a BBN for assessing human-autonomy collaboration (HAC) performance is presented. This BBN combines factors related to the human operators and AMSs that influence HAC performance. The most important factors are the human operators' experience, human operators' training, and workload. The influence of the human operators on the collaborative performance is mediated by the level of autonomy of the AMSs. Autonomous function reliability and the situational awareness capabilities of the AMSs are the most influential factors on HAC performance pertaining to AMSs.

This thesis also presents a process for developing safety indicators for the operation of AMSs. Safety indicators can be used to monitor the level of safety during the operation of AMSs. To prevent the occurrence of accidents, the proposed process allows the development of an indicator system that enables the human operators to assess whether the level of risk of operation is increasing. The indicators address subsystems and aspects of the organisation that allow the identification of organisational and technical weaknesses that may lead to an accident if not controlled.

Software governs the AMSs and controls most of the AMSs during operation. The software needs to be safe and reliable. The human operators have a supervisory role and need to act when AMSs are not capable of coping with the situation any longer. The risk modelling processes, approaches, and aspects that are described in this thesis address the need to ensure and demonstrate that AMSs are safe with respect to relevant human, technical, and organisational factors. Therefore, the implications from the risk-influencing factors identified for HAC are important for the design of human-machine interfaces and control systems to keep the human operators aware of the situation and enable them to act when required.

Acknowledgements

This work would not have been possible without the people who accompanied me during the doctoral research period. I would like to thank these people here.

First, I would like to express my gratitude to Professor Ingrid Bouwer Utne, who gave me the opportunity to conduct this research. She provided guidance, advice, and discussion when needed. I would also like to thank my co-supervisors, Professor Ingrid Schjølberg and Professor Ali Mosleh. Both provided insightful discussions and asked the right questions when needed. I would like to thank Professor Mosleh for the opportunity to visit UCLA and conduct research there in the spring of 2017.

I would like to acknowledge the financial support provided by the Department of Marine Technology (IMT) at NTNU, which sponsored the research. I was affiliated with NTNU AMOS, and I appreciate this opportunity. The *Norges tekniske høgskoles fond* provided financial support for my research visit to UCLA, and this is acknowledged.

I would like to thank Professor Stein Haugen from IMT at NTNU for his advice and contribution as co-author to the first article. I also want to thank Marilia Abilio Ramos and Xue Yang for the good collaboration and research, where I could contribute as a co-author. Some of the research enclosed in this thesis was conducted in collaboration with the Advanced Underwater Robotics Laboratory at NTNU, which provided data and allowed me to participate on several cruises with R/V Gunnerus. In particular, I would like to thank Professor Martin Ludvigsen, Petter Norgren, and the crew of NTNU's research vessel R/V Gunnerus.

The research period would have not been the same without the people at the Department of Marine Technology. I would like to thank my colleagues at IMT and AMOS for all the discussions, and activities related to research and, more commonly, not related to research. The coffee and lunch breaks were something to look forward to each day. I would like to thank my former office mates: Jeevith Hegde, who is also co-author on two of the articles included in this thesis, and Erlend Kvinge Jørgensen

Thanks go to my friends, both in Trondheim, Oslo, and the rest of the world. You made it a pleasant time providing company and diversion when needed. Thanks, in particular, go to my flatmate Cristian, with whom I could share the good sides and hardships of PhD life.

Lastly, I would like to thank my family for their love and support in all my decisions regarding my career and providing necessary help and aid when needed. Thanks and sincere gratitude go to Lilia, who supported me during all the good and hard times, calmed me down, made me laugh, and opened my eyes to things that I otherwise would never have discovered.

Content

Table of figures

Table of tables

Abbreviations

AAWA	Advanced autonomous waterborne applications
AMOS	Centre for autonomous marine operations and systems
AMS	Autonomous marine system
AROV	Autonomous remotely operated vehicle
AUR Lab	Autonomous underwater robotics laboratory
AUV	Autonomous underwater vehicle
BBN	Bayesian belief network
CAS	Collision avoidance system
CSRM	Context-based risk assessment method
DNV-GL	Det Norske Veritas and Germanischer Lloyd
DFM	Dynamic flowgraph model
DP	Dynamic positioning
EN	European standard
ETA	Event tree analysis
FMEA	Failure mode and effect analysis
FT	Fault tree
FTA	Fault tree analysis
HAC	Human-autonomy collaboration
HMI	Human-machine interface
IAEA	International atomic energy agency
IEC	International Electrotechnical Committee
IEEE	Institute of Electrical and Electronics Engineers
IMT	Department of Marine Technology at NTNU
ISO	International Organization for Standardization
LoA	Level of autonomy
MASS	Maritime autonomous surface ship
MUNIN	Maritime unmanned navigation through intelligence in networks
NASA	National Aeronautics and Space Administration
NS	Norwegian standard
NTNU	Norwegian University of Science and Technology in Trondheim
OECD	Organization for Economic Co-operation and Development
PhD	Philosophiae doctor
RIF	Risk-influencing factor
RO	Research objective
ROV	Remotely operated vehicle
RQ	Research question
SFF	Centre of excellence (Norwegian: Senter for fremragende forskning)

SCC	Shore control centre
SPAR-H	Standardised plant analysis risk-human reliability analysis
STPA	System-theoretic process analysis
UCLA	University of California, Los Angeles
USV	Unmanned surface vehicles
UUV	Unmanned underwater vehicle

Thesis Structure

This thesis is written in the form of a collection of articles. The first part presents an introduction to the research questions, research objectives, and a summary of the research executed to address these. Part II contains the publications that form the basis of this thesis. The articles present the methods, results, discussions, and conclusions in detail. Part III lists all previously completed theses at IMT.

Part I is structured as follows. Section 1 introduces the topic of AMSs and the challenges with respect to risk and safety of these. Furthermore, Section 1.1 states the research question and the research objectives underlying the research work. Section 1.2 summarizes the delimitations of the conducted research.

Section 2 presents the theoretical background for the research work, defining risk and associated concepts. In addition, this section gives an overview on the state-of-the-art risk assessment and analysis for AMSs and software systems.

Section 3 summarizes the research methodology. It answers the questions: How the research presented in the articles was approached and how the research in the articles is related.

Section 4 describes how the conducted research addresses the research objectives and research questions. It gives an overview on the methods, results, and discussion of the contribution of the articles included in this thesis.

Section 5 concludes the executed research, highlighting the implications and contributions for academic research and for the industry. Section 5.3 gives an overview on research areas that should be addressed in the future.

This page is intentionally left blank

Publications

The following articles are included in Part II of this thesis. Articles 1 through 5 are journal articles and Article 6 is a conference article. They provide detailed information on the results and contributions that are presented in Part I.

Article 1

Thieme, C. A., Utne, I. B. & Haugen, S. 2018. *Assessing ship risk model applicability to marine autonomous surface ships, Ocean Engineering*, 165, pp. 140-145, DOI: 10.1016/j.oceaneng.2018.07.040.

Article included in Part II, pp. 63-80.

Article 2

Thieme, C. A., Mosleh, A., Utne, I. B. & Hegde, J. Submitted. *Incorporating software failure in risk analysis – Part 1: Software functional failure mode classification. Submitted for review to Reliability Engineering and System Safety*.

Submitted article included in Part II, pp. 81-112.

Article 3

Thieme, C. A., Mosleh, A., Utne, I. B. & Hegde, J. Submitted. *Incorporating software failure in risk analysis – Part 2: Risk analysis process and case study. Submitted for review to Reliability Engineering and System Safety*.

Submitted article included in Part II, pp. 113-148.

Article 4

Thieme, C. A. & Utne, I. B. 2017. *A risk model for autonomous marine systems and operation focusing on human-autonomy collaboration. Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability,* 231, pp. 446-464, DOI: 10.1177/1748006x17709377

Article included in Part II, pp. 149-170.

Article 5

Thieme, C. A. & Utne, I. B. 2017. *Safety performance monitoring of autonomous marine systems*. *Reliability Engineering & System Safety,* 159*,* March, pp. 264-275, DOI: 10.1016/j.ress.2016.11.024

Article included in Part II, pp. 171-184.

Article 6

Thieme, C. A., Utne, I. B. & Schjølberg, I. 2015. *A risk management framework for unmanned underwater vehicles focusing on human and organizational factors.* Proceedings of the ASME 2015 34th International Conference on Ocean, Offshore and Arctic Engineering OMAE2015, 31.05.-05.06.2015. St. John's, NL, Canada. ASME.

Article included in Part II, pp. 185-194.

Declaration of Authorship

Table 1 summarises the contributions of each author to the publications included in this thesis. Each author is assigned letters according to their contribution to the respective article. The letters correspond to the following contributions:

A. Initial research idea and concept;
B. Data collection;
C. Data analysis;
D. Writing and design of the draft of the article;
E. Critical review of the article.

In general, all authors have participated in the writing, critical review, and approval of the final version of each article. In many cases, the authors participated in data analysis or collection, which qualifies each author as an author according to the Vancouver recommendations for authorship by the International Committee of Medical Journal Editors (2018).

Table 1 Contribution of each author to the publications enclosed in this thesis.

Author	Article					
	1	2	3	4	5	6
Christoph A. Thieme	A-E	A-E	A-E	A-E	A-E	A-E
Ingrid B. Utne	A, C-E	D, E	D, E	A, D, E	A, D, E	A, C-E
Ingrid Schjølberg						A, D, E
Ali Mosleh		A, D, E	A, D, E			
Stein Haugen	C-E					
Jeevith Hegde		B, D, E	B-E			

Publications not Included in this Thesis

During the doctoral research period, two more publications were produced. These are not part of this thesis.

Article 7

Thieme, C. A., Utne, I. B. & Schjølberg, I. 2015. *Risk modeling of autonomous underwater vehicle operation focusing on the human operator. In:* PODOFILLINI, L., SUDRET, B., STOJADINOVIC, B., ZIO, E. & KRÖGER, W., eds. 25th European Safety and Reliability Conference, ESREL 2015, 7.-10.09.2015. 2015 Zürich, Switzerland. Boca Raton, London, New York, Leiden: CRC Press, Taylor & Francis Group, pp. 3653–3660.

Article 8

Yang, X., Utne, I. B. & Thieme, C. A. 2018. *A preliminary review of hazard identification techniques for autonomous operations in Norwegian aquaculture.* Probabilistic Safety Assessment & Management conference PSAM 14, Los Angeles, CA, USA.

This page is intentionally left blank

Part I – Main Report

This page is intentionally left blank

1 Introduction

The marine environment is harsh and vast. Operating on the seas is demanding for seafarers and operators of equipment for marine operations. The exposure to this environment is considered to be one of the most dangerous to work in, resulting in several thousand accidents with multiple fatalities each year (Allianz Global Corporate & Speciality, 2018). High costs are associated with damages to the environment and loss of assets resulting from marine accidents (ibid.). Most of these accidents are attributed to human error (ibid.). Manning ships with a sufficient number of personnel is expensive, and accommodation areas for the crew use space that could be used for transporting cargo and payload.

In addition, the world is facing the challenge of global warming, fuelled by the emissions of greenhouse gases, such as carbon dioxide, sulphur oxides, and nitrous oxides. The global maritime shipping industry is responsible for 2.5% of all global greenhouse gas emissions, with a predicted increase of 50% to 250% until 2050 (European Commission, 2018). This creates additional pressure on the maritime industry, which is supposed to reduce these emissions and use more environmentally sustainable technologies.

Autonomous systems, such as self-driving cars and self-controlled flying drones, are being developed and prototypes are in use. Similarly, autonomous marine systems (AMSs) are being developed to reduce the exposure of personnel to the environment. These will reduce the risk for crew and operators and will allow different operational concepts, such as slow steaming, to save money and reduce the emissions of shipping. AMSs are expected to reduce risk and cost with respect to personnel significantly. Especially for long-lasting science missions, synergy effects from the deployment of AMSs are to be expected, increasing the operational range and capabilities. Figure 1 summarizes the different types of AMSs.

Autonomous ships, so-called maritime autonomous surface ships (MASSs; Rødseth and Nordahl, 2017) are considered the future of maritime transportation. Especially for transport in coastal areas, they are expected to reduce the amount of trucks on roads and thereby improve the regional traffic and economic situations. The first MASSs are expected to operate soon in coastal shipping in the fjords of Norway (Kongsberg Maritime, 2017). Autonomous ferries (DNV-GL, 2018) and offshore supply vessels (Kongsberg Maritime, 2016) are to be expected in operation soon. Several projects aim at developing concepts for MASSs and establishing a base for standardisation, for example, ReVolt (DNV-GL, 2015;

Tvete, 2015), Maritime Unmanned Navigation through Intelligence in Networks (MUNIN, 2012), or Advanced Autonomous Waterborne Applications (AAWA, 2016).

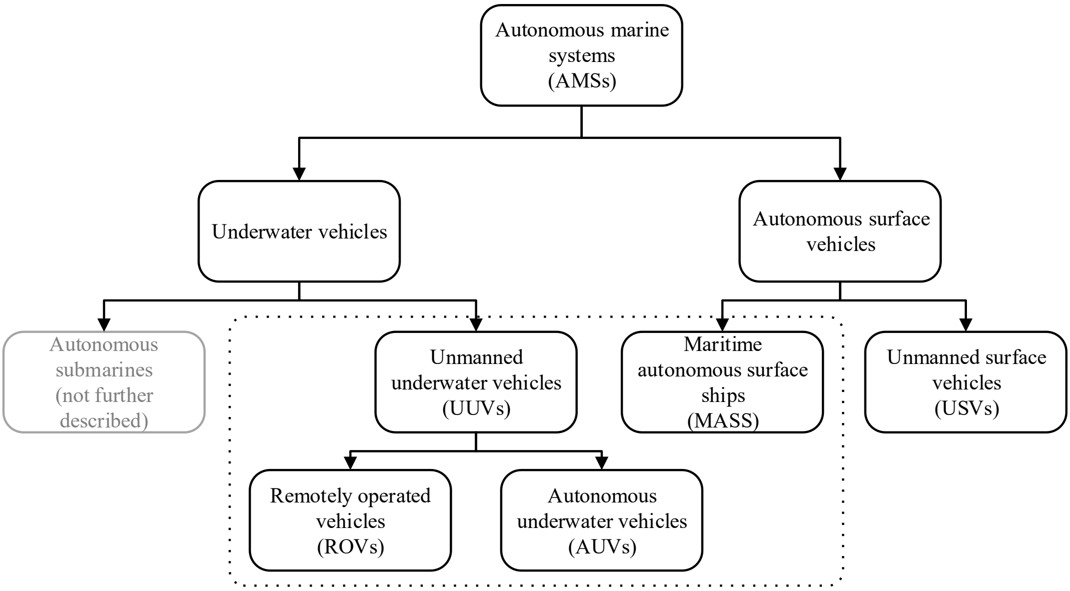

Autonomous marine systems (AMSs)
Underwater vehicles
Autonomous surface vehicles
Autonomous submarines (not further described)
Unmanned underwater vehicles (UUVs)
Maritime autonomous surface ships (MASS)
Unmanned surface vehicles (USVs)
Remotely operated vehicles (ROVs)
Autonomous underwater vehicles (AUVs)

Figure 1 Types and classifications of autonomous marine systems, adapted and extended from Rødseth and Nordahl (2017). The dotted box marks the systems that were investigated in this thesis.

Unmanned surface vehicles (USVs) are small vessels (2–15 m length and 1.5–10 t weight; Bertram, 2008) that are remotely controlled. They are used for surveys of the oceans and do not transport goods or people. Concepts of MASSs and USVs have received increased attention in recent years due to the technical feasibility. The first prototypes are in use (Manley, 2008; Yan et al., 2010; Bertram, 2008, 2016). The Trondheimsfjord in Norway became one of the first test areas for MASSs and USVs (Norwegian Maritime Authority, 2016).

Underwater vehicles, especially unmanned underwater vehicles (UUVs), are commonly used AMSs. Autonomous underwater vehicles (AUVs) and remotely operated vehicles (ROVs) make up this category. AUVs have existed for several decades and are characterised through their capability to survey the subsea ocean environment on a larger scale than with divers or submarines. AUVs are used, for example, for mapping the seafloor, inspecting pipelines, or measuring sea water properties of the water column (Yuh et al., 2011). AUVs do not need input from human operators under normal operation conditions. AUVs operate in conditions similar to missions in space, with little prior knowledge and high uncertainty (Harris et al., 2016).

ROVs are underwater robots, which are normally controlled by human operators through a tethered connection to a surface vessel to land-based human operators (Christ and Wernli, 2007) or a subsea garage. ROVs with more autonomous functionalities are developed. ROVs will become so-called autonomous ROVs (AROVs). AROVs will be used in underwater intervention, maintenance, and repair operations to reduce operational cost. Large parts of an operation should be carried out by AROVs without human operators intervening (Hegde et al., 2015; Hegde, 2018). Hence, AUVs and AROVs will be more difficult to differentiate in the future. Manned autonomous submarines have not been discussed yet but might be relevant in the future, for example, as tourist attractions. Hence, these systems are not further discussed.

AMS must be safe and reliable to be accepted by the regulatory bodies and the public (Nautilus Federation, 2018; Earthy and Lützhöft, 2018). The public may demand that AMSs have a better safety performance than conventional ships. For this purpose, it is necessary to demonstrate that these systems will not lead to an increased level of risk, in particular, with respect to the loss of life, damage to the environment, or damage to assets (Wróbel et al., 2017; Utne et al., 2017).

A risk-based approach was recommended to be part of the future international legislation for MASSs (Danish Maritime Authority, 2018). Lloyd's Register (2016) requires that a risk-based design approach is used for the development of MASSs (called cyber-enabled ships by Lloyd’s Register). For operation of MASSs in the Trondheimsfjord in Norway, actors need to demonstrate that the risk was assessed and evaluated as reasonably low (Norwegian Maritime Authority, 2016).

This thesis addresses risk modelling and risk analysis of AMSs. The next section defines the research questions and objectives that underlie this thesis.

1.1 Research Questions and Objectives

The study by the Nautilus Federation (2018) shows that a high scepticism towards risk assessments for MASSs is to be expected. In general, risk assessments of new technological systems are difficult since there has been no experience with these systems and hence no data for evaluation are available. In addition, complex system interactions are to be expected for AMSs due to the technical complexity of the system operating in the marine environment (Harris et al., 2016; Utne et al., 2017).

There are three main challenges with respect to the risk assessment of complex AMSs. First, inclusion of software in risk assessment of technological systems is difficult (Mosleh, 2014). The software that is used in autonomous ships could introduce significant risks for the company operating AMSs (Earthy and Lützhöft, 2018).

The second challenge lies in assessing the role of the human operators, who will supervise the AMS and intervene if necessary. Different levels of autonomy (LoA) influence the collaborative performance of the human operators with the AMSs (Cummings, 2014). This interaction needs to be included in risk assessments to fully address the associated implications (Utne et al., 2017; Earthy and Lützhöft, 2018).

The third challenge with respect to risk assessment of AMSs arises from the dynamic marine environment. The weather, sea state, and environment are changing continuously. The AMS will encounter different traffic situations and interact with different marine stakeholders. The changes in environmental, technical, and organisational conditions occur more frequent than updates of risk assessments (Knegtering and Pasman, 2013). Hence, tools are necessary to monitor the level of risk of AMSs' operations. Therefore, risk monitoring of AMS operation is an important aspect to ensure safe operation.

This thesis addresses this challenge of analysing and modelling the risk of AMSs. It attempts to answer the overall research question, *how to model and analyse the risk of autonomous marine systems*, and contribute to safe operation of AMSs by answering three research questions. The first research question aims at the identification of risk analysis needs for AMSs. Based on the first research question, two research objectives have been formulated.

Overall Research Question:
How to model and analyse the risk of autonomous marine systems?

Research Question 1 (RQ1):
Are current risk assessment methods and models able to assess the level of risk of autonomous marine systems?

The first research objective (RO1) aims at identifying and assessing risk-influencing factors (RIFs) that will actually influence the level of risk of AMSs. The RIFs are related to how AMSs are different from conventional maritime and marine systems, such as ships or submarines.

With the input from RO1, the second research objective (RO2) aims at identifying whether the current models and methods are suitable for the assessment and modelling of the level of risk of AMSs. If they are suitable, it is possible to also identify the shortcomings of these models or the necessary modifications to these models.

Research Objective 1 (RO1):
Derive risk-influencing factors that need to be included in risk assessments for autonomous marine systems.

Research Objective 2 (RO2):
Review current risk assessment methods for marine systems and assess their applicability to autonomous marine systems.

The second research question investigates how software and the interaction between the AMS and the human operators affect the operation of AMSs. To visualise and collect the information, which is obtained through RO1 and RO2, risk models are developed in research objective 3 (RO3).

Research Question 2 (RQ2):
How do software and human interaction with the system contribute to the level of risk of autonomous marine systems?

Research Objective 3 (RO3):
Develop models for the assessment of the influence of software and human operators on the risk level of the operation of autonomous marine systems.

With the risk models at hand, one question arises for the operation of AMSs, research question 3 (RQ3), which addresses how the level of risk can be monitored. As a way to answer the research question, research objective 4 (RO4) was formulated, which uses the collected information from the other research objectives.

Research Question 3 (RQ3):
How can the level of risk of autonomous marine systems be monitored during operation with respect to their specific system requirements?

Research Objective 4 (RO4):
Develop a risk monitoring approach based on safety indicators for autonomous marine systems.

1.2 Delimitations

There are several types of AMSs and concepts. To limit the extent of the research, the focus in this thesis is on MASSs and AUVs. MASSs are still under development or in the concept stage. Unlike conventional ships, insufficient data are available for MASSs. Hence, quantification of models is difficult. The work in this thesis is mainly qualitative in nature.

For quantitative examples related to AUVs, the data and experience from the Applied Underwater Robotics Laboratory (AUR Lab) at NTNU and data from the literature have been used. The MASS concepts that are described in this thesis were developed based on available information in the literature. Future AMSs might be operated differently than described in this thesis. AUVs and MASSs are different concepts, and the application of results obtained from the analysis of AUVs must be considered with care when being transferred to other AMSs. However, the results are assumed to be generally valid, and the transfer of knowledge is assumed to be possible.

AROVs are used in the case study in Articles 2 and 3. A decision-support system is analysed that was developed for AROVs. A similar system may be envisioned for MASS or AUV operation. Other results may be transferred to the case of operation of AROVs.

Occupational risk and the hazards for personnel working in the maritime environment are not covered in this thesis. The thesis aims at addressing major accidents that will lead to loss of the AMSs or to severe damage to assets, the environment, or people. Interaction of third-party individuals that find a lost AMS may also lead to damage to their health (Stokey et al., 1999). However, this aspect is not further considered.

In general, the results described in Section 4 may apply to other autonomous systems, such as other AMSs, autonomous cars, autonomous aircrafts, or autonomous spacecraft. The application and transfer of results should be executed with care, and adaption may be necessary to fit the context of operation of these systems.

2 Theoretical Background

This section summarises the background for this thesis. It addresses risk assessment, risk modelling, and sets these in the context of AMSs. This section also summarises previous work on risk analysis of AMSs, how software has been included in risk analysis, and provides a brief introduction on safety indicators for safety monitoring.

Risk may be interpreted in different ways (Aven, 2012). Hence, the risk concept that is used throughout this thesis must be clarified. Risk is the *effect of uncertainty on objectives* (International Organization for Standardization (ISO), 2009). It can be further refined as a combination of the *consequences of an event and the associated likelihood of the event* (ISO, 2009). The measure of likelihood may be probability or frequency. Risk may therefore be defined as the combined answer to *(i) what can go wrong, (ii) how likely is it that it will happen, and (iii) if it does happen, what are the consequences?* (Kaplan and Garrick, 1981).

Risk assessment is the process to find answers to these questions. It consist of *risk identification, risk analysis,* and *risk evaluation* (ISO, 2009). Risk analysis is the *process to comprehend the nature of risk and to determine the level of risk* (ISO, 2009). A *source of danger that may cause harm to an asset* (Rausand, 2011) is called a hazard. Reviewing hazards may identify sources of potential harm to the system, which gives input to risk analysis. A RIF is *an aspect (event or condition) of a system or an activity that affects the risk level of this system or activity* (Øien, 2001b).

Risk management of an organisation comprises the c*oordinated activities to direct and control an organi*sation *with regard to risk* (ISO, 2009). The *set of components that provide the foundations and organi*sational *arrangements for designing, implementing, monitoring, reviewing and continually improving risk management throughout the organi*sation (ISO, 2009) form the risk management framework.

The aim of risk management is to reduce the risk associated with an activity to an acceptable level. For this purpose, risk-mitigating measures are identified based on the findings in the risk analysis and evaluation process. Risk-mitigating measures, or so-called barriers (Sklet, 2006),may be, among others, engineering solutions that modify or enhance the design of a system, procedures, or specialised training for human operators.

With these concepts defined, the risk associated with the operation of AMSs can be identified and assessed. For MASSs, risk analysis attempts to find the likelihood of events, such as collision, allision, grounding, or stranding and the assessment of the severity of the associated consequences, such as damage to people, damage of the MASS, damage to other ships and infrastructure, pollution of the environment, loss of cargo, or damage to cargo (Kretschmann, Rødseth, Tjora, et al., 2015). Fire, loss of hull integrity, and loss of stability are also system hazards that may be a result of the aforementioned events (ibid.). Collision and grounding of a conventional vessel is the largest contributor to the level of risk of conventional maritime shipping (Pedersen, 2010). The risk spectrum for MASSs might change compared to conventional ships (Wróbel et al., 2016).

For AUVs, the hazards and the associated consequences are different. An AUV may, among others, collide with fixed structures, the seafloor, vessels, and other swimming objects; lose integrity or power; or stop actuating. This may lead to the loss of the vehicle, damage to equipment, damage to other assets, or loss of data (Manley, 2007; Utne and Schjølberg, 2014; Brito and Griffiths, 2016a). In most cases, these consequences are related to monetary loss.

2.1 Autonomous Marine Systems and Related Concepts

Autonomy is the ability of a system to make its own decisions and to adapt to the circumstances to achieve the overall goal of the system. This is achieved without additional decisions or input from supervising agents, such as human operators or other systems (Vagia et al., 2016). Autonomy and automation are often used interchangeable (ibid.). However, these two concepts are not the same. Automation means that a task that has been executed formerly by a human is executed by a technical system instead. An autonomous system is automated. However, an automated system is not necessarily autonomous. The concept of autonomy goes further than simply substituting human operators with a technological system. Several more dimensions need to be considered (cf. Huang, 2007; Insaurralde and Lane, 2012; Kaber, 2017), when defining the autonomy of a system.

Levels of automation, often called LoAs, refer to the degree of automation. This implies a certain degree of independent decision making to achieve an overall mission goal from operators for a higher LoA (ibid.). This thesis focuses on autonomy. An autonomous system that is capable of changing the LoA according to the circumstances is designed with adaptive autonomy (Sheridan, 2011).

Scales that are often used to describe the LoA were presented by Sheridan and Verplank (1978) and Endsley and Kaber (1999). More detailed reviews of LoA scales can be found in the work by Insaurralde and Lane (2012), or Vagia et al. (2016). Rødseth and Nordahl (2017) presented LoA scales for continuously manned and periodically or completely unmanned MASSs. Utne et al. (2017) presented LoAs that address generally autonomous systems but are well suited for AMSs. This scale is described in Table 2.

Table 2 Levels of autonomy developed specifically for autonomous marine systems, adapted from Utne et al. (2017).

LoA	Type of operation	Description
1	Automatic operation (remote control)	The AMS operates automatically; the human operators give high-level mission plans and control each phase. Mission and environmental data are presented through a human-machine interface (HMI) to the human operators.
2	Management by consent	The AMS makes recommendations to the human operators, suggesting mission or process-related actions for specific functions. The AMS prompts the human operators for decision or information at critical or important mission points. Such an AMS may have a limited bandwidth for communication due to the distance to the operational base. The AMS may also act independently from the human operators for a period of time if delegated to do so.
3	Semi-autonomous operation or management by consent	The AMS takes its own decisions when the required reaction time for human operators is too short. The human operators have the possibility to change certain parameters and cancel or redirect certain actions within a certain time frame. The human operators are specifically alerted and called upon just for certain exceptions and decisions.
4	Highly autonomous operation	The AMS carries out a mission or a process without input from human operators. It can plan and re-plan its actions to achieve the mission or execute the process. The human operators may gain information on the progress, but the AMS operates independently and intelligently in an often unstructured environment.

Rødseth and Nordahl (2017) showed that unmanned is not the same as autonomous. A MASS may still be manned, while the bridge is unmanned part of the time. An unmanned ship may be remotely controlled (e.g., an USV), which may be located in LoA 1 of the scale. It is expected that different MASS concepts will emerge, addressing different LoAs and using different operational concepts, depending on the application of the MASS (Rødseth and Burmeister, 2015).

Three main concepts are currently differentiated for MASSs, (i) low manned vessels with a partly unattended bridge (Bertram, 2016; Rødseth and Nordahl, 2017), (ii) a swarm of MASSs supervised by one manned ship, also called master-slave (Bertram, 2016), and (iii) MASSs supervised by shore control centres (SCCs; Rødseth et al., 2014; Rødseth and Nordahl, 2017). A MASS with low manning (i) is an intermediate solution to unmanned autonomous vessels during the transition period (Bertram, 2016; Kongsberg Maritime, 2017). The crew on board a vessel is then reduced in comparison to conventional shipping.

AUVs are currently located in LoA 2 and LoA 3 of the scale presented above. They are pre-programmed and will execute their mission only with limited input from the human operators. ROVs are typically found in LoA 1, whereas AROVs may be found in LoA 2 and LoA 3 (Hegde et al., 2018). Current ships employ advanced technological systems, such as complex automation and dynamic positioning (DP) systems (Utne et al., 2017; Earthy and Lützhöft, 2018). The DP system aboard a vessel is used to keep a vessel on a certain position on the sea within a small allowable tolerance. It enables a vessel to manoeuvre very precisely and hence may be an important part of a MASS (AAWA, 2016; Bureau Veritas, 2017).

2.2 Risk Assessment of Autonomous Marine Systems

2.2.1 Autonomous Underwater Vehicles and Autonomous Remotely Operated Vehicles

Risk assessment and modelling of AUVs and AROVs were presented to some extent in the literature. These use operational data. Few risk assessments for MASSs are available and only for conceptual MASSs. Hence, this subsection summarises the available literature on AUVs and AROVs first. The literature on MASSs and advanced technological ships is summarised in the latter part of this section.

Griffiths and Brito (2008) presented a risk management approach for AUVs used in Polar Regions. The approach was based on an expert assessment of the RIFs that affect the risk of operation. Risk was defined as the loss of an AUV in combination with the probability of loss. They developed a Bayesian belief network (BBN) that includes RIFs related to the environmental conditions, the vessel from which the operation is conducted, and the ability of the AUV to operate in these conditions.

Brito and Griffiths (2009) used an expert's judgement on the fault logs of an AUV to assess the mission risk of polar expeditions. The risk estimation was used to justify the mission executions, which would not have been conducted without the risk assessment. Hence, they argued that risk management is a suitable tool to make decisions related to AUV missions. Brito et al. (2010) demonstrated how this knowledge may be transferred and used in risk management and decision making for polar missions.

Griffiths et al. (2009) used expert elicitation on the fault logs of two REMUS 100 AUVs. The elicitation assessed the probability of loss, depending on the length of the operation, for different scenarios. The results may be used to limit the mission length based on the acceptable risk level.

Utne and Schjølberg (2014) described a systematic risk assessment approach for AUVs. The process followed the generic risk management standard described by the International Organization for Standardization in ISO 31000 (2009). Along with the process, they identified and developed potential hazardous events and a hazard taxonomy for AUV operations in the categories of natural events, technical events, human behaviour events, and malicious events. The hazard taxonomy may be considered when conducting a risk assessment for AUVs. They described how safety systems for AUVs may be developed and guided by the standard for functional safety electrical/electronic/programmable electronic safety-related systems, under IEC 61508 (2010).

Brito and Griffiths (2016a) summarised risk management of AUVs and how this may assist in operational decision making. They highlighted that different stakeholders may have a different perspective on the risk of AUVs that must be addressed, for example, the owner of the AUVs may focus on the loss of the system, while the users may define risk through the unavailability of the AUVs.

Brito and Griffiths (2016b) extended their previous work by including more detailed RIFs in their BBN. These are RIFs, such as underwater obstacles, surface conditions, ice coverage, and vessel recovery effectiveness. They demonstrated how the model may be used to assess the risk retrospectively and to predict the risk of a mission using encountered and expected conditions, respectively. They (ibid.) also showed how performing missions may be used to update the probabilities in the BBN to refine the risk estimates with operational experience.

Harris et al. (2016) addressed the challenges of risk assessment of AUVs. They reviewed methods that are used to assess the risk of operating AUVs. As predictive tools, they listed failure mode and effects analysis (FMEA), fault tree analysis (FTA), and event tree analysis (ETA). The lack of reliability data for components may be overcome by simulation approaches, expert predictions, data recording, and consequent data updating. They (ibid.) addressed the challenge of operating multiple vehicles and vehicle types together. They recommended addressing the different levels of the system, for example, the subsystem, vehicle, or multiple vehicles, to identify the risk contributors through a bottom-up approach.

Hegde et al. (2016) developed a safety indicator approach for AROVs. The indicators measure time to collision, mean time to collision, and mean impact energy. These indicators are associated with RIFs, namely, acceleration, distance to target, vehicle velocity, and drag. The mission may be divided into different phases to estimate the level of risk during these different phases and provide decision support.

Brito and Griffiths (2018) demonstrated how, similar to the processes presented in their earlier research, the effect of risk mitigation measures and fault removal attempts on the mission risk can be assessed through expert elicitation.

Hegde et al. (2018) presented a BBN for assessing the probability of preliminary aborting an AROV mission. They included mainly technical nodes, representing the subsystems involved in the operation, the subsea environment, and organisational nodes, such as LoA, human supervisor state and training, and other operational parameters. They used expert judgement with industry professionals for the quantification of the network. The BBN provides decision support to human AROV operators.

2.2.2 Maritime Autonomous Surface Ships and Advanced Ships

Only a few publications focus on MASSs. Rødseth and Tjora (2014) and Rødseth and Burmeister (2015) presented their approaches to assess the risk of MASSs in an early development phase as part of the MUNIN project. They highlighted that such an assessment should be conducted before the system requirements are defined. The process should give input to the requirements. Initial scenarios were used to identify hazards, develop risk-mitigating measures, and verify the design.

Rødseth and Burmeister (2015) listed hazards for MASSs that need the most attention: interaction with other ships, errors in detection and classification of obstacles, breakdown of propulsion, the behaviour of the MASSs in heavy weather, and issues related to cyber security and piracy. They reasoned that human operators, located in a SCC, may communicate with other traffic participants or identify objects through a screen, to solve hazardous situations.

Kretschmann, Rødseth, Tjora, et al. (2015) and Kretschmann, Rødseth, Sage Fuller, et al. (2015) presented the qualitative and quantitative evaluation, respectively, of the MUNIN project, including a risk assessment. For the qualitative evaluation, they conducted a hazard identification, which named, among others, human error in operation and maintenance, foundering in heavy weather, and cyber security issues as the most important hazards. The quantitative risk analysis was based on the results by Jensen (2015).

Jensen (2015) used ETA and FTA to assess the level of risk of the MUNIN concept cargo ship during a voyage. For the assessment of the initiating events (i.e., possible collision encounter and groundings) automatic identification system data were used. The ETA and FTA contained mainly events that are related to the environment and technical failures. Software and human operators' failures were incorporated with a low level of detain in the risk model.

The AAWA (2016) report summarised, among others, safety and security related considerations of the AAWA project. The authors highlighted some issues that need attention with respect to their contribution to the level of risk of MASS operation:

- Reliability of safety critical equipment;
- Validation of safety related information and communication equipment, and software;
- Reliability of mechanical equipment (i.e., maintenance);
- Remote operation and monitoring of the MASS by human operators;
- Security aspects, such as cyber security;
- Management of emergencies with remotely located human operators.

Li et al. (2016) presented their experimental and simulation results on the roll behaviour of an USV. The knowledge gained through these tests provided input on the risk analysis of USVs in heavy sea and adverse weather. For this particular USV design, the safe regions of operation could be identified. These may be included in the control system of USVs to operate them safely in adverse sea states.

Wróbel et al. (2016) presented their results for the hazard analysis of MASSs. They used a BBN to structure their findings in four groups:

- Navigation, which may lead to grounding or collision;
- Engineering (steering, propulsion, and electrical power);
- Stability and associated considerations;
- Miscellaneous (e.g., fire, piracy, and communication).

These groups are influenced by root causes, which were summarised as maintenance regime, sensor performance, control algorithms, external information, and alerting (of human operators on shore).

Utne et al. (2017) addressed risk management of manned and unmanned AMSs, with different LoAs. The article highlighted the RIFs that affect the risk of MASSs, which are grouped in mission/operation, environment, and system. In mission/operation and system, they mentioned human fatigue, human absence from the control room, human-operator intoxication, and human-operator training and experience. They recommended developing risk models that provide online decision support, incorporating the identified RIFs to ensure safe operation.

Wróbel et al. (2017) conducted a what-if analysis to assess the effect of introducing autonomy to ships in the maritime industry. MASSs may contribute to the reduction of the frequency of collisions and groundings. The main challenge was considered the remoteness of the human operators, which has the benefit that the risk to personnel is reduced. However, this remoteness implies that, in case of an accident, the human operators cannot take a recovering role. The uncertainty with respect to the operational conditions was highlighted. The results were preliminary, and more knowledge about the operational conditions and hazards that affect MASSs is needed.

Rokseth et al. (2016, 2017) used the system-theoretic process analysis (STPA) and FMEA to identify, analyse and develop verification goals for the DP system of ships. The use of the FMEA and STPA together revealed hazards that would be overlooked using just one of these methods. Consequently, hazard and risk analysis were improved, which may lead to improved verification goals and may improve systems.

Ait Allal et al. (2017) presented their considerations for safe and reliable communication architectures for MASSs for different areas with different communication infrastructures. The research emphasised that there should be a failsafe strategy, in case the primary communication line to the SCC or other marine participants is not available.

Earthy and Lützhöft (2018) summarised the challenges of MASSs and advanced ships with respect to demonstrating safety and compliance with regulations. They highlighted that it is necessary to demonstrate that it is safe to reduce the manning level. An important aspect for this demonstration is the assessment of the interaction between operators and the crew with the MASS or advanced ship. Risk assessments and the included factors depend highly on the concept of use and the concept of the ship.

Valdez Banda and Kannos (2018) analysed the hazards for an autonomous city ferry project in Finland through STPA with the input from different stakeholders and experts. They identify 15 hazards, which may if not controlled lead to accidents, such as collision, grounding, or passengers being involved in accidents. One of the focus areas is software failures of artificial intelligence.

Wróbel et al. (2018) developed a safety control structure model for MASSs. They analysed it with the STPA to identify possible scenarios in which control structures may become inadequate. The analysis highlighted its preliminary status, addressing the uncertainty with respect to the design of MASSs. Technical issues have been identified as the factor contributing most to safety-related issues, followed by the interaction between SCC and the regulatory framework it needs to act under.

Human operators are relevant during the remote control of MASSs. Otherwise, they play a minor role as contributors to the risk of the MASS operation (ibid.). However, they may take a supervisory role and may need to take (limited possible) actions in case of a technical failure. The environment does not pose a hazard since MASSs must be able to cope with the circumstances (ibid.). Safety control functions need to be implemented on different levels of the whole MASS system, ranging from the regulatory framework over organisational issues to the technical solution of the MASS itself.

From the presented literature, it can be seen that the risk contribution from the operation software is very important. However, only a few publications assess or address the risk contribution from the software system in an AMS. None attempted a detailed analysis or presented methods to attempt such a venture. In addition, the risk contribution from the interaction of human operators with the AMS may affect the level of risk significantly. However, this has not been addressed in the literature in detail. The next subsection summarises approaches to analyse the contribution of software to the level of risk of a system or an operation.

2.3 Software Contribution to the Level of Risk

Software is and will be an important part of AMSs. Software is found in sensors, control systems, guidance and navigation systems, and monitoring systems. Remote control and supervision will be accomplished through human-machine interfaces (HMI) on shore or on board the vessel. Software fails mainly due to design error, and unlike hardware systems, the failure rate of software is not time dependent (Chu et al., 2009). No attempts were made so far to include the software contribution in the risk level in risk assessments of AMSs. Hence, this section summarises the findings from other types of systems' analyses.

The reliability of software is of high concern, and many models for software reliability exist. For examples of references and models, see Chu et al. (2010), Yamada (2014), or the recommended practices on software reliability (Institute of Electrical and Electronics Engineers (IEEE), 2016). However, software that works reliably does not need to work safely and may contribute under certain circumstances to the level of risk (Garrett and Apostolakis, 1999). Therefore, software reliability methods are only applicable to a limited extent to assess the contribution to the risk level.

Software FMEA has been in use for many years, but no formal process has been developed (Ozarin, 2003). Several publications described software FMEA, for example, Ristord and Esmenjaud (2002), Huang et al. (2009), Stadler and Seidl (2013), Park et al. (2014), and Prasanna et al. (2014). Li et al. (2003) and Li (2004) developed a failure mode taxonomy for software focusing on the functions.

The Organization for Economic Co-operation and Development (OECD, 2014) presented a taxonomy for hardware and software failure modes. The taxonomy builds on research by Li et al. (2003); Li (2004); Li et al. (2005), Authen et al. (2010), Authen and Holmberg (2012, 2013), and Holmberg et al. (2012), among others. One challenge with the variety of taxonomies of software failure modes is the inconsistency in the system level of application. Only OECD (2014) attempted to clearly state the system levels that are addressed, for example, the overall system level, division level, instrumentation and control unit level, and instrumentation and control categories.

Garrett et al. (1995) and Guarro et al. (1996) developed the dynamic flowgraph methodology (DFM) to assess the dependability and safety of software systems in two steps: (i) build the model for the software system and (ii) analyse the model to build fault trees (FT). A DFM model is a directed graph with functional relations (the causality network) and conditions that trigger functional relations (conditional network). The software system is considered a flow of information that is manipulated by different software functions.

Several extensions have been developed for the DFM, such as those by Al-Dabbagh (2009) and Al-Dabbagh and Lu (2010), who developed reusable models to describe networked control systems. Aldemir et al. (2009) and Aldemir et al. (2010) use Markov cell mapping in combination with DFM. The process is capable of capturing system behaviour dynamically, discovering event sequences that otherwise would not have been found.

The US National Aeronautics and Space Administration (NASA) implemented DFM in their context-based software risk assessment methodology (CSRM), which is used to identify, analyse, and mitigate risks associated with space missions (Stamatelatos et al., 2011; Guarro et al., 2013). For simple software systems, NASA suggested using simple logic models (e.g., FTA). For complex software systems that depend on timing, NASA suggested using DFM.

Hewett and Seker (2005) assessed the contribution of embedded software systems similar to DFM. Decision tables represent the software behaviour. Timed FTs are built through backwards reasoning.

Li et al. (2003) and Li (2004) used the developed software failure mode taxonomy to implement identified failure modes in FTA and event sequence diagrams. Only selected failure modes were implemented in the analysis, not differentiating the levels of software decomposition.

Wei (2006) and Wei et al. (2010) built on the failure modes by Li et al. (2003) and Li (2004) and described the propagation behaviour through a software system for these failure modes. This behaviour is used to simulate the software behaviour in the case of software failure and to implement the resulting relevant events in risk analysis.

Zhu (2005) and Zhu et al. (2007) included software failures in dynamic risk assessment, building on the work by Wei (2006). Software failures are *injected* in a dynamic risk analysis model and the simulation *reacts* to these failures. The resulting software behaviour is implemented in dynamic FTs.

Leveson (2004) and Leveson et al. (2012) described STPA. In the process, a model of the system is built and assessed from a control perspective to identify hazards that arise through insufficient control. Abdulkhaleq and Wagner (2015) and Abdulkhaleq et al. (2015) extended the STPA for automated model checking of critical software applications, identifying potential hazardous situations from a software model and verifying that the control actions are safe.

Most of the described methods require that the software system is fully specified or even exists. The AMSs, especially MASSs, are conceptual, the information on the software may be limited due to an early development stage or proprietary reasons. Hence, approaches are needed that enable risk analysts to identify software-related failure events that may influence the level of risk and include these in risk assessment.

2.4 Risk Monitoring and Safety Indicators

Industries, which are associated with a high level of risk, such as the chemical process industry or the oil and gas industry, monitor the risk with indicators on different organisational levels, for example, an industrial level (e.g., Vinnem, 2010), at a company level (e.g., Reiman and Pietikainen, 2012), or at a single plant or unit of operation (e.g., Skogdalen et al., 2011; Hassan and Khan, 2012; Øien, 2013). These indicators are specific to the level of organisation and have only limited applicability for other levels.

Risk and safety indicators are not the same and have different implications. Risk indicators are founded on a risk-based approach, such as a risk model (Øien et al., 2011), for example, Øien (2001a) or Øien (2001b). A risk indicator is the measurable variable of a RIF. Safety indicators, on the other hand, represent how safe a system presently is, for example, through event indicators, barrier indicators, activity indicators, and programmatic indicators (Øien, 2013). A safety indicator may be defined as a measurable or operational variable that can be used to describe the level of safety of operation (adapted from the definition of an indicator by Øien, 2001b). A thorough review of safety indicators was presented by Swuste et al. (2016), who discussed other definitions that are in use in the scientific community and in different industries.

Occupational safety indicators and process safety indicators are the two main types of safety indicators in use. The first type of indicators may not apply to AMSs since they refer to the wellbeing of personnel and accidents related to personnel. In many cases of AMS operation, personnel will be working away from the AMS. Early warning indicators provide information on the performance of barriers, which can prevent a potential incident (Øien, 2008). Outcome indicators measure the occurrence of undesired events, reflecting actual operational safety performance (International Atomic Energy Agency (IAEA), 2000). For AMSs, such undesired events could be the loss of position, navigational errors, or misinterpretation of sensor data.

Rødseth et al. (2014) described a performance monitoring approach for the MUNIN project. They did not address the safety indicator directly. However, they suggested indicators in the categories of *functional condition index*, *functional status index*, *technical status index*, and *technical condition index*. These indicators aim at highlighting different conditions of the MASSs. Some of the identified indicators could be described as safety indicators, such as *high traffic density*, *reduced manoeuvrability*, or *reduced redundancy* capabilities.

Only one publication addressed risk monitoring or AMSs. This indicates that more work is necessary to develop approaches for risk and safety monitoring of AMSs.

3 Research Approach

3.1 Research Methodology

Research can be categorised into two types, basic research and applied research (Roll-Hansen, 2009). Basic research attempts to manage and increase knowledge that is generally valid. Applied research can be understood as the intersection between science and politics. It is dedicated to solving a practical problem by applying general knowledge (ibid.). Both types are not mutually exclusive, and basic research can solve a practical problem. The research that is described in this doctoral thesis can be classified primarily as applied research. Knowledge is applied to solve the challenges associated with risk analysis of AMSs. To achieve this goal, some knowledge had to be developed.

Three main types of research methodology can be differentiated: qualitative, quantitative, and mixed methods (Creswell, 2014). Qualitative research uses an inductive approach to analyse and interpret data. It aims at understanding a particular topic and extending it to a general context. Quantitative research uses measurable variables to analyse relationships and support outlined theories. This is an inductive approach since theories might be rejected, and other explanations can be found through the analysis (ibid.).

Mixed research approaches combine qualitative and quantitative methods to analyse a research problem in more detail and provide a more complete understanding. These concepts are not as clearly distinguishable as the definition suggests. The research approaches always have an overlap and share common aspects (Creswell, 2014).

In addition to the research methodology types mentioned above, Kothari (2004) named several other research types. Some of these are briefly described. Descriptive research is defining and describing the state of a system or an object to document its state and circumstances. Analytical research analyses a system by examining system variables and exploring the research topic in this way.

The thought process that aims to develop and formulate theories and concepts drives conceptual research. Empirically driven research uses data analysis to establish and formulate theories and information. Inferential research builds on knowledge in a database to derive characteristics and relationships. Experimental research executes more control on data since data are obtained directly in specifically designed experiments. A simulation approach to research derives data from an artificial environment (i.e., a numerical model; ibid.).

The articles included in this thesis fall into different categories of the research methodology types. Table 3 summarizes the research methodology types that can be found in the articles enclosed in this thesis. As stated earlier, the research is predominantly applied research. Hence, basic and applied research types are not listed.

Table 3 Research types found in the articles enclosed in this thesis.

Research type	Article 1	Article 2	Article 3	Article 4	Article 5	Article 6
Qualitative	Yes	Yes	Yes	No	No	No
Quantitative	No	No	No	No	No	No
Mixed	No	No	No	Yes	Yes	Yes
Descriptive	Yes	Yes	Yes	Yes	Yes	No
Analytical	Yes	No	No	Yes	Yes	Yes
Conceptual	Yes	Yes	Yes	Yes	Yes	Yes
Empirical	No	No	No	No	No	No
Inferential	Yes	Yes	Yes	Yes	Yes	Yes
Experimental	No	No	Yes	No	No	Yes
Simulation	No	No	No	No	No	No

Article 1 presents a literature review and hence is qualitative. The research is inferential for the same reasons. The research is also conceptual since the analysis relies on the conceptual description of MASSs. The research in Article 1 is descriptive and analytical, describing the state-of-the-art MASSs and analysing existing risk models for ships with respect to their applicability to MASSs.

Article 2 merges failure mode taxonomies for software functions in a comprehensive failure mode taxonomy. The work is qualitative and is based on the description of the derived taxonomy and case study. The article infers knowledge from early publications using conceptual considerations and descriptions.

Article 3 uses the failure mode taxonomy from Article 2 and describes the propagation behaviour of the failure modes through a software system. This makes it descriptive and qualitative research. The findings are inferred from the reviewed literature. The failure modes and their propagation behaviour are embedded in a risk assessment process that is demonstrated on a conceptual case study.

Article 4 develops a BBN for assessing the collaborative performance of human operators and AMSs (i.e., AUVs). The article presents mixed research since most of the information is gathered qualitatively. However, quantification is attempted. The model is useable across different systems and types of operation. For this purpose, the research is both descriptive, describing the case study object, and analytical, for building the model. The model is generated through inferential research from the literature in similar areas of research.

Article 5 presents a safety indicator development approach for AMS and demonstrates its applicability. The development of the safety indicator approach is qualitative. Quantitative results are presented for the case study. Therefore, it is a mixed research article. For the same reasons, the article is descriptive with respect to the safety indicator process and is analytical with respect to the case study applying the safety indicators. The process in the article is conceptual and has not been applied empirically. The case study is inferential since all data are gained from existing documentation, and no additional experiments or simulations were carried out.

Article 6 presents a risk management framework, which is exemplified by a quantitative case study. The article uses qualitative and quantitative methods. The article is analytical since it presents a risk analysis. This analysis is conceptual. Data are gained through literature and expert judgement, which makes the article both inferential and experimental.

3.2 Research Work Process

The PhD project and the resulting research can be divided into three phases (i.e., familiarisation, addressing research objectives, and summarising research). Figure 2 summarizes the activities during the doctoral research project phases. Arrows indicate iterations and connections between different activities within each phase. The interaction between individual activities of different phases are not depicted for better readability.

The first phase was a familiarisation phase. The research questions were defined initially. These formed the basis for identifying suitable courses that were taken to fulfil NTNU's requirements for attaining a PhD degree. Four courses were selected, which covered different aspects of risk modelling and management.

Concurrently with the courses, the literature was reviewed and summarised. The knowledge from the courses was applied together with the findings from the summarized literature and developed into Article 6. This was not a linear process. Several iterations of the literature review, summarising results and refining research questions and objectives, were conducted.

In the second phase, the research questions and objectives were addressed specifically. Research needs were identified through the objectives. These were addressed through the development and adaption of methods and processes for the specific research objectives. The research findings were summarised in research articles, which were submitted to scientific journals. These are Articles 1 through 5. In addition, the participation in conferences and workshops provided input to the research activities.

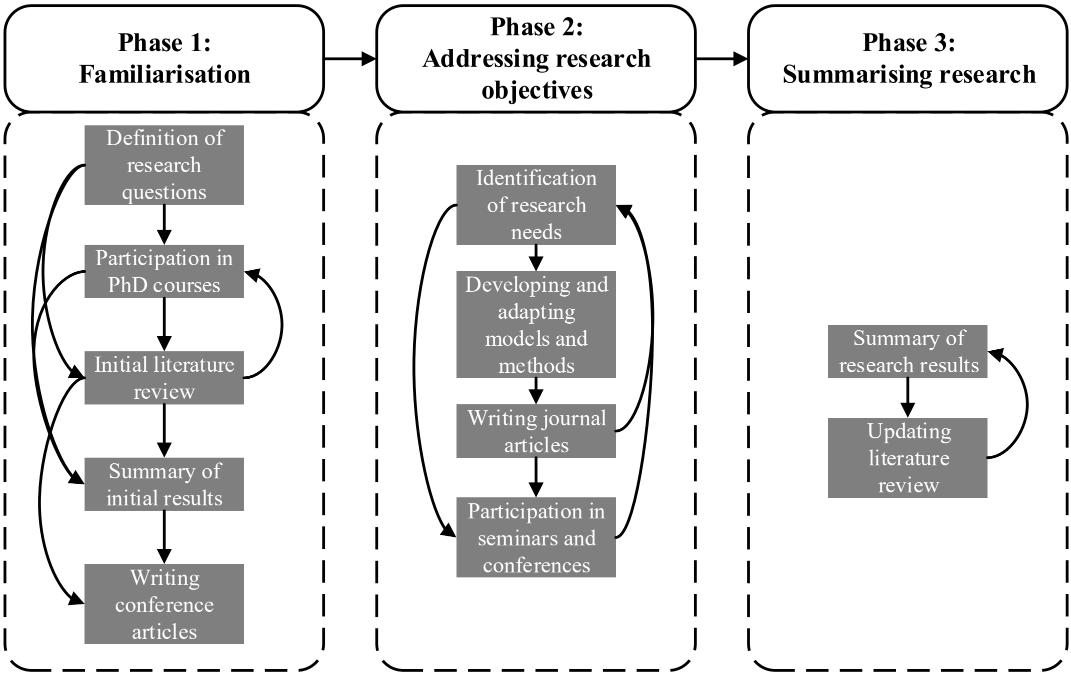
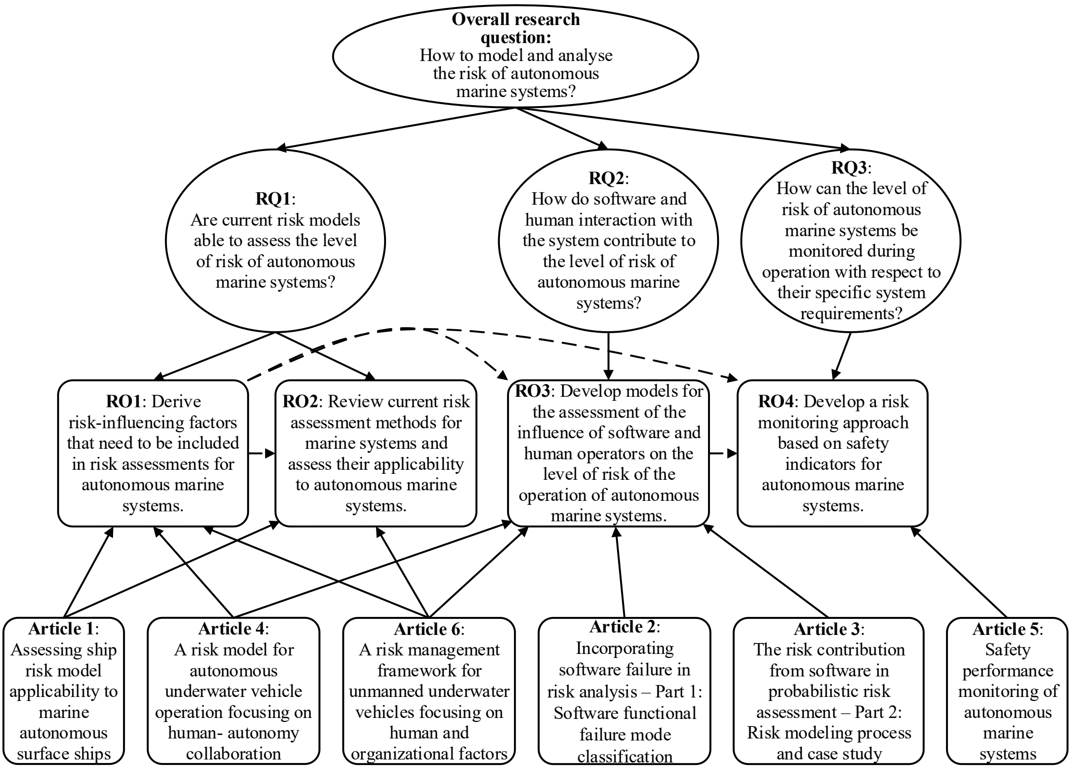

Phase 1: Familiarisation
Definition of research questions
Participation in PhD courses
Initial literature review
Summary of initial results
Writing conference articles
Phase 2: Addressing research objectives
Identification of research needs
Developing and adapting models and methods
Writing journal articles
Participation in seminars and conferences
Phase 3: Summarising research
Summary of research results
Updating literature review

Figure 2 Work process followed in the course of the doctoral studies.

The third phase concludes the PhD research project by summarising the work in this thesis. For that purpose, the conducted research was reviewed and evaluated against the research objectives and questions. The literature review was updated and forms the background and state of the art in this thesis.

4 Results and Contributions

This section summarises the contributions from the articles to the research objectives. Figure 3 depicts the relationship between the research questions, research objectives, and articles. Most articles address several of the research objectives. Only Article 5 addresses RO4. Findings from the research objectives that are used to address other research objectives are represented through broken lines. Findings from RO1 are used in addressing RO2, RO3, and RO4. Findings from RO3 are used in RO4.



Overall research question: How to model and analyse the risk of autonomous marine systems?
RQ1: Are current risk models able to assess the level of risk of autonomous marine systems?
RQ2: How do software and human interaction with the system contribute to the level of risk of autonomous marine systems?
RQ3: How can the level of risk of autonomous marine systems be monitored during operation with respect to their specific system requirements?
RO1: Derive risk-influencing factors that need to be included in risk assessments for autonomous marine systems.
RO2: Review current risk assessment methods for marine systems and assess their applicability to autonomous marine systems.
RO3: Develop models for the assessment of the influence of software and human operators on the level of risk of the operation of autonomous marine systems.
RO4: Develop a risk monitoring approach based on safety indicators for autonomous marine systems.
Article 1: Assessing ship risk model applicability to marine autonomous surface ships
Article 4: A risk model for autonomous underwater vehicle operation focusing on human- autonomy collaboration
Article 6: A risk management framework for unmanned underwater vehicles focusing on human and organizational factors
Article 2: Incorporating software failure in risk analysis – Part 1: Software functional failure mode classification
Article 3: The risk contribution from software in probabilistic risk assessment – Part 2: Risk modeling process and case study
Article 5: Safety performance monitoring of autonomous marine systems

Figure 3 Research questions and objectives in relation to the articles included in this thesis. Broken lines represent knowledge gained through one research objective that is used further for other research objectives.

4.1 Contribution to Research Objective 1

The first research objective is to derive RIFs that need to be included in risk assessments for AMSs. This objective is addressed mainly through Articles 1, 4, and 6.

Article 1 uses nine criteria to assess current ship risk models for their applicability to MASSs. These criteria represent considerations and RIFs that highlight the main differences between operation of MASSs and conventional ships. The criteria are derived through a systems engineering process.

For the identification of the criteria, the operation of conventional vessels and MASSs are described and compared. Through a need analysis, requirements for MASSs are identified. The requirements are used to formulate nine criteria that represent considerations that need to be included in risk models for MASSs. These are summarised in Table 4.

Table 4 Criteria that summarise the risk-influencing factors that need to be considered in a maritime autonomous surface ship risk model. Reproduced from Thieme et al. (2018).

#	**Criteria**
C1	Inclusion of software and control algorithm performance
C2	Inclusion of human-machine interfaces and ergonomic considerations
C3	Inclusion of communication between vessels and shore base
C4	Inclusion of communication between human operators
C5	Inclusion of aspects of maintenance and reliability of system performance
C6	Inclusion of functional redundancy
C7	Consideration of different operational modes and change of level of autonomy
C8	Inclusion of communication between human operators and other marine participants
C9	Consideration of different crew levels

The most significant difference and the most challenging aspect of risk assessment of MASSs, compared to conventional ships, is the software system that controls the MASSs. The performance of the software and algorithms needs to be included as contributing to the level of risk of MASS operation (C1).

All monitoring, controlling, and most of the communication will be executed through HMI. Communication and HMI refer to Criteria 2, 3, 4, and 8. Since MASSs may be unmanned, the maintenance policy and the implications on the system reliability need to be considered for MASSs (C5). Different operational concepts should be considered for modelling the risk of MASSs during a voyage (i.e., C7 and C9).

Functional redundancy needs to be considered (cf. Criterion 6). A MASS may have different systems that fulfil the same function, or a function carried out autonomously may be also carried out by human operators. These different circumstances need to be reflected in a risk model. Although the human operators may be removed from the ship and may supervise it remotely, they might need to take control of the MASS, communicate with each other to solve a situation, communicate with other stakeholders through a remote connection, or take other actions.

Article 6 highlights the need to include human operator-related RIFs in risk assessment of UUVs. In the case study on an AUV, the human operator actions are modelled through the standardised plant analysis risk model human reliability analysis (SPAR-H; Gertman et al., 2005; Whaley et al., 2011). The SPAR-H method considers performance-shaping factors for the assessment of human error probability. These are available time, stress, stressors, experience, training, complexity, ergonomics, procedures, fitness for duty, and work processes.

These performance-shaping factors are relevant for the operation of AMSs; therefore, this method was chosen. However, the assessment of human reliability with the SPAR-H method does not completely capture important RIFs, such as the LoA. Hence, a more detailed analysis of RIFs related to the human operators that are relevant for the operation of AMSs is conducted.

Article 4 presents the findings from this analysis. The article identifies the RIFs that are relevant for human-autonomy collaboration (HAC), which is defined as the joint performance of the human operator and the autonomous system during a mission of an AUV, its deployment, or its retrieval. A literature study on risk assessment of autonomous and highly automated systems and on RIFs that may influence HAC forms the background for the RIFs.

Table 5 summarizes the identified RIFs. The RIFs that influence the HAC performance can be summarised as human-operator-related RIFs, mission-dependent RIFs, and technical RIFs. Relevant human-operator-related RIFs include, among others, *communication*, *operators' experience and training*, and *trust*. Mission-dependent RIFs are, for example, *mission duration* and *number of vehicles per operator*. Technical RIFs depend on the type of AMS, for example, *etiquette*, *false alarm rate*, *reliability of autonomous functions*, and *time delay of transmission*. The interaction between the RIFs is further described and discussed in Section 4.3.2, where the RIFs are included in a BBN.

Table 5 Identified risk-influencing factors that influence the collaborative performance of human operators with an autonomous system. Reproduced and adapted from Thieme and Utne (2017a).

Identified risk-influencing factor	**Description**
Communication	Information exchange between human operators to fulfil the assigned mission.
Etiquette	*Set of prescribed and proscribed behaviours that permits meaning and intent to be ascribed to actions* (Parasuraman and Miller, 2004) of the system.
False alarm rate	Rate of status messages that contain erroneous information.
Fatigue	*Inability [of the operators] to function at the desired level due to incomplete recovery from the demands of prior work and other waking activities* (Gander et al., 2011).
Feedback from the system	Factor summarising the way a system gives feedback to the human operators on the status, intentions, and actions.
Interface design	Design principles applied to the physical and virtual interfaces of the system.
Level of Autonomy	The degree of the system ability to make independent decisions. This depends on the type of operation to be carried out and type of AUV. This relationship is not further included in the model.
Mission duration	The duration of use and operation of AUVs for a mission. It also depends on the type of mission, type of vehicle, and environmental conditions. These interactions are not modelled since they would require that environmental and technical aspects are fully included in the model.
Number of vehicles per operator	Number of AUVs and AUV types that one human operator operates concurrently.
Operators' experience	Level of experience of the operators with operation of the AUVs. This includes experience with AUV programming, AUV maintenance, AUV deployment and recovery, assessment of the marine environment, and working in the marine environment.
Operators' training	The amount of relevant training human operators received for operation of AUVs. Relevant training includes training with respect to AUV programming, AUV deployment and recovery, AUV maintenance, the marine operation environment, and working in the marine environment.
Procedures	Provided documentation that prescribes operation and provides guidance to human operators.
Reaction time	Time the human operators need to react to a situation that needs their attention.
Reliability of autonomous functions	The system ability to perform its functions as required during the time of use. This includes mission-relevant and diagnostic functions.
Shift scheme	Pattern that determines the human-operator working and resting time.
Situation awareness of human operators	Perception and comprehension of the AUV state and situation during operation by the human operators, and projection of the future state.
Situation awareness of vehicles	The vehicle ability to perceive information, interpret, integrate, and assess relevance of that information and to predict the future with this information and prior background knowledge.
Task load	Number of tasks that must be executed concurrently by one human operator. This evaluation should include the consideration of the complexity of the tasks.
Time delay of transmission	Time that a message needs to transmit from the AUV to the human operators or vice versa.
Trust	*Users' willingness to believe information from a system or make use of it.* (Parasuraman and Miller, 2004)
Workload	The work demand encountered by the human operators during AUV operation.

Discussion

The criteria used in Article 1 are derived from a high-level system and need analysis of MASS concepts. Hence, they address a wide range of issues. The criteria do not identify detailed RIFs that need to be considered in risk assessments and models for AMSs. This may be attributed to the fact that several concepts of MASSs and types of AMSs exist. The criteria are used as the basis for an overall assessment of ship risk models in Section 4.2, which exhibit different levels of detail.

Article 6 highlights that the human-operator-related RIFs may have a significant influence on the mission outcome of UUV operation. A risk assessment should generally consider human-operator-related RIFs, where applicable. The SPAR-H was developed for the nuclear industry for human-operator tasks. This may make it unsuitable for the direct application to AMSs and, in the case of Article 6, for UUV. Hence, more investigation into the interactions between human-operator-related RIFs and other RIFs was necessary. This research resulted in Article 4.

Relevant RIFs that were identified in Article 4 were derived from the literature on human interaction with highly automated systems and unmanned vehicles. It is believed that all these RIFs are relevant for human-autonomous system collaboration. Some RIFs found in the literature have been excluded from the list, such as the perceived risk associated with a specific task (e.g., Parasuraman and Riley,1997; Lee and See, 2004; Parasuraman and Miller, 2004; Sheridan and Parasuraman, 2005). Most AUV cannot be controlled remotely, so this RIF was excluded from the considerations. For MASSs, however, the risk associated with a certain task might lead human operators to not use an autonomous functionality. Hence, this RIF should be considered in MASS risk models.

Additionally, RIFs related to the environment, acting on AMSs or on the human operators have not been analysed. Such RIFs are well studied and included in most existing risk models for conventional ships and AUV (cf. Article 1; Brito et al., 2010; Brito and Griffiths, 2016b). Environmental RIFs that affect the human operators in a control room for AMSs should also be included. This research topic is addressed in ergonomics (Karwowski, 2006), which is not a research topic that exclusively applies to AMSs. Ergonomics is not addressed further.

4.2 Contribution to Research Objective 2

The second research objective is to review current risk assessment methods for marine systems and assess their applicability to AMSs. This objective is addressed mainly through Articles 1 and 6.

Article 1 reviews allision, collision, grounding, and stranding risk models for conventional ships and assesses them against the set of criteria presented above. A literature review forms the basis of this article. The analysis identifies gaps in the current risk models for conventional ships and points out those modelling approaches that are promising for MASSs. The analysis considers vessels during transit, excluding vessels carrying out a special operation (i.e., fishing). The analysis focuses on the qualitative modelling. The quantification of risk models for MASSs needs to be assessed for MASSs and cannot be adopted from existing risk models for conventional ships.

The review considers publications published since 2005, which present models for assessing the probability of allision, collision, grounding, and stranding for ships. For this purpose, 64 relevant publications with relevant models are identified. These models are assessed against the above-described nine criteria. Ten models fulfilled at least six criteria. None of the models reviewed are ready for direct adoption, and additional work is required to adapt them for risk analysis of MASSs.

None of the models fulfilled all criteria. Three models were developed specifically for risk analysis of MASSs and fulfilled most criteria. None of the analysed models presented a detailed assessment of software risk or failure of software-based systems. At most, the models included modelling elements, such as failure of navigational equipment, failure of the control system, or software error. However, all the analysed models provide insight in how certain aspects of the ship operation are currently included in risk models. For example, several models use BBN as a modelling technique, which should be considered for risk modelling of MASSs.

Article 6 reveals that human-operator-related RIFs are not explicitly considered in risk models for UUVs. Experience from other industries shows that highly automated, autonomous, or unmanned systems still can be subject to human error, which influences the operational performance and may increase the level of risk for certain operations. Examples can be found in the supervision of autonomous systems (e.g., Bainbridge, 1983; Sheridan, 2006; Parasuraman and Wickens, 2008), in the remote control of robotic vehicles (e.g., Chen et al., 2007; Chen et al., 2011), and in control of underwater robot operation (e.g., Sheridan and Verplank, 1978; Sheridan, 1982; Chellali and Baizid, 2011).

Discussion

Article 1 assesses the risk models for conventional ships with a focus on their applicability to MASSs. Only models that are used for the frequency or probability assessment are reviewed. Both consequences and the quantification of the models are specific to the purpose of the model and to the vessel being analysed. Hence, these were not assessed since any risk model needs to be developed for its purpose.

Due to the number of publications and the generality of the criteria, it is not possible to assess each of the risk models individually. In addition, most of the models do not present a high level of detail in the model description to assess the included RIFs and model structures in more detail.

With respect to Article 6, the literature shows that RIFs in relation to human operators are not considered for UUV operation and especially AUV operation. Similarly, they have not been the focus of MASS risk models so far due to the immaturity of the systems. Human-operator-related RIFs need sufficient attention to fully reflect the operational concepts adopted for AMSs.

4.3 Contribution to Research Objective 3

The third research objective is to develop models for the assessment of the influence of software and human operators on the risk level of the operation of AMSs. This objective is addressed through Articles 2, 3, 4, and 6. The contribution to this research objective is two-fold. First, a process has been developed to incorporate software in risk analysis (Section 4.3.1). Second, RIFs related to human operators have been included in risk models for AMSs (Section 4.3.2).

4.3.1 Integration of Software in Risk Models

Articles 2 and 3 are accompanying articles, which present the research conducted on the incorporation of software in risk analysis. For this purpose, a process based on the identification of failure modes for the functions of a software is chosen. A failure mode is the manner an item or system fails (IEC EN, 2006). Software failure modes are context specific (Li, 2004); a failure mode may lead to negative consequences in one scenario, whereas the same failure mode may not have any effect in another chain of events.

As was shown in Section 2.3, several approaches to assess the risk contribution from software use failure modes. One challenge with existing software failure mode taxonomies is that they do not adhere to one level of analysis. Only OECD (2014) attempts to define failure mode taxonomies, which are applicable to specific levels of analysis.

For the analysis of software failure modes, a functional view on the software is chosen. This has the advantage of facilitating the definition of functional requirements (EN, 2004). A functional view is used in several risk assessment methods (Chu et al., 2009). From a functional perspective, the system can be analysed from the early design phases without the complete system being available. Software can be decomposed into its functions. Decomposing it in several iterations will lead to the software code level. The code level is not of concern for the developed process.

Figure 4 presents the process that follows the guidelines for risk management of ISO 31000 (2009) to incorporate the software failure modes in risk analysis. Steps 2, 3, and 4 have been adapted for this purpose. The individual main contribution of each article is marked in the figure.

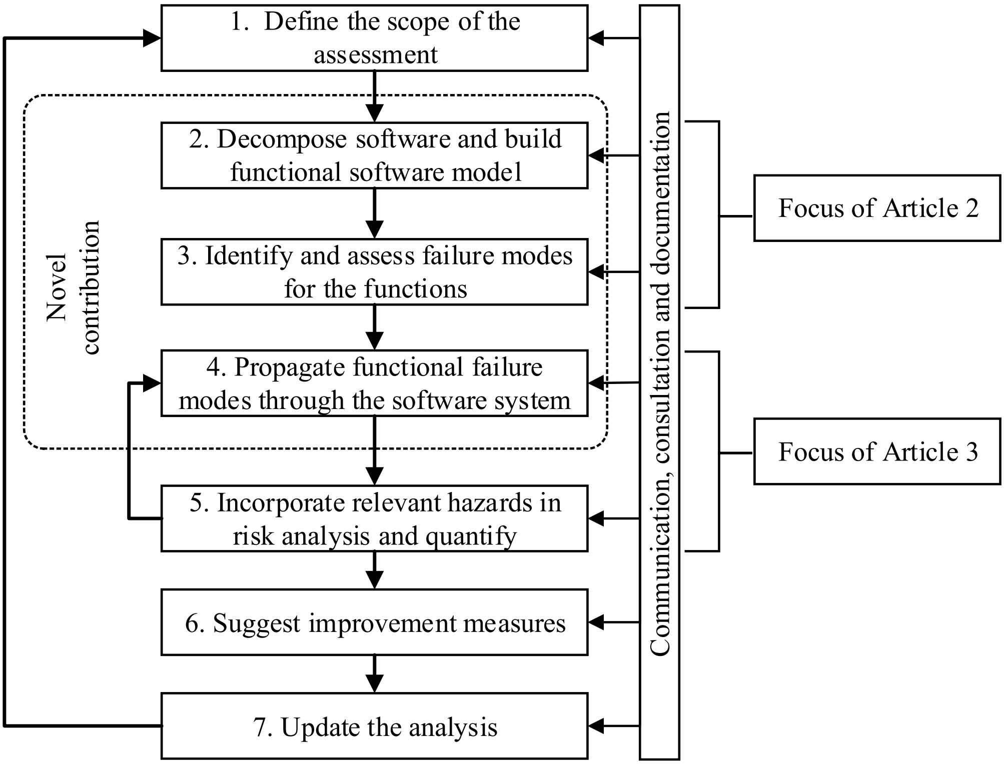

1. Define the scope of the assessment
2. Decompose software and build functional software model
3. Identify and assess failure modes for the functions
4. Propagate functional failure modes through the software system
5. Incorporate relevant hazards in risk analysis and quantify
6. Suggest improvement measures
7. Update the analysis
Novel contribution
Communication, consultation and documentation
Focus of Article 2
Focus of Article 3

Figure 4 Steps for risk assessment incorporating software and its influence on the level of risk. The contributions from Articles 2 and 3 are highlighted. Adapted from Thieme et al. (submitted-a).

The first step of the process is to define the scope of the analysis, including the context of use and the level of detail of the analysis. This information is necessary for the failure mode identification, propagation, and definition of the context for the risk analysis.

A case study on an underwater collision avoidance system (CAS), presented in Hegde (2018), and Hegde et al. (submitted), exemplifies the process. The underwater CAS is a support system for AROV operation and visualises objects that are within collision range and the position and orientation of the ROV. The case study is not explained in more detail in this thesis. The reader is referred to the articles for more information.

In the second step, the software is decomposed. Figure 5 shows the adopted view on software functions and how different types of failure modes can be applied to the different elements of a function. The process section is where the functional behaviour and computations are executed, turning inputs into outputs. Function failure modes are associated with this part of the function.

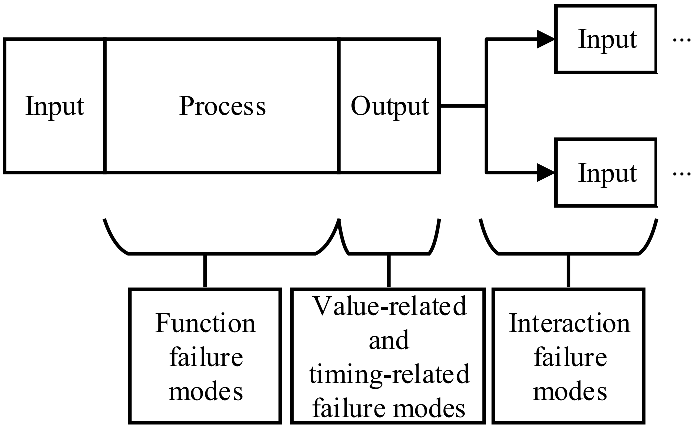

Input
Process
Output
Input
...
Input
...
Function failure modes
Value-related and timing-related failure modes
Interaction failure modes

Figure 5 View of a software function and associated software failure modes. Reproduced from Thieme et al. (submitted-b).

A function has at least one output, which may be a numerical value, binary value, or functional call. Each function has one or several inputs. Value-related and timing-related failure modes are associated with the output part of a software function. Software functions are executed in a required order, as they are executed on demand or periodically. Each function passes on information to other functions or calls another function. These interactions between the functions, represented through arrows, are associated with interaction failure modes. Software and its functions might interact with external interfaces. External interfaces are systems, such as other software systems, sensors, databases, or human operators through HMI.

The information from the functional decomposition of the software is used to build a functional software model. The functional software model assists in the identification of failure modes (Step 3) and analysis of the propagation of the failure modes through the software system (Step 4) to identify the effect on the external interfaces. These effects may be incorporated in the risk analysis (Step 5).

An example for a functional software model, the AROV underwater CAS case study, is shown in Figure 6. Rectangles represent the functions, and circles represent the external interfaces. These are connected via two types of connectors. Continuous lines represent the exchange of information or data between these elements. The dotted line represents functional dependencies, such as function calls or the execution loop. The broken line visualises the software boundary, separating function blocks from the external interfaces.

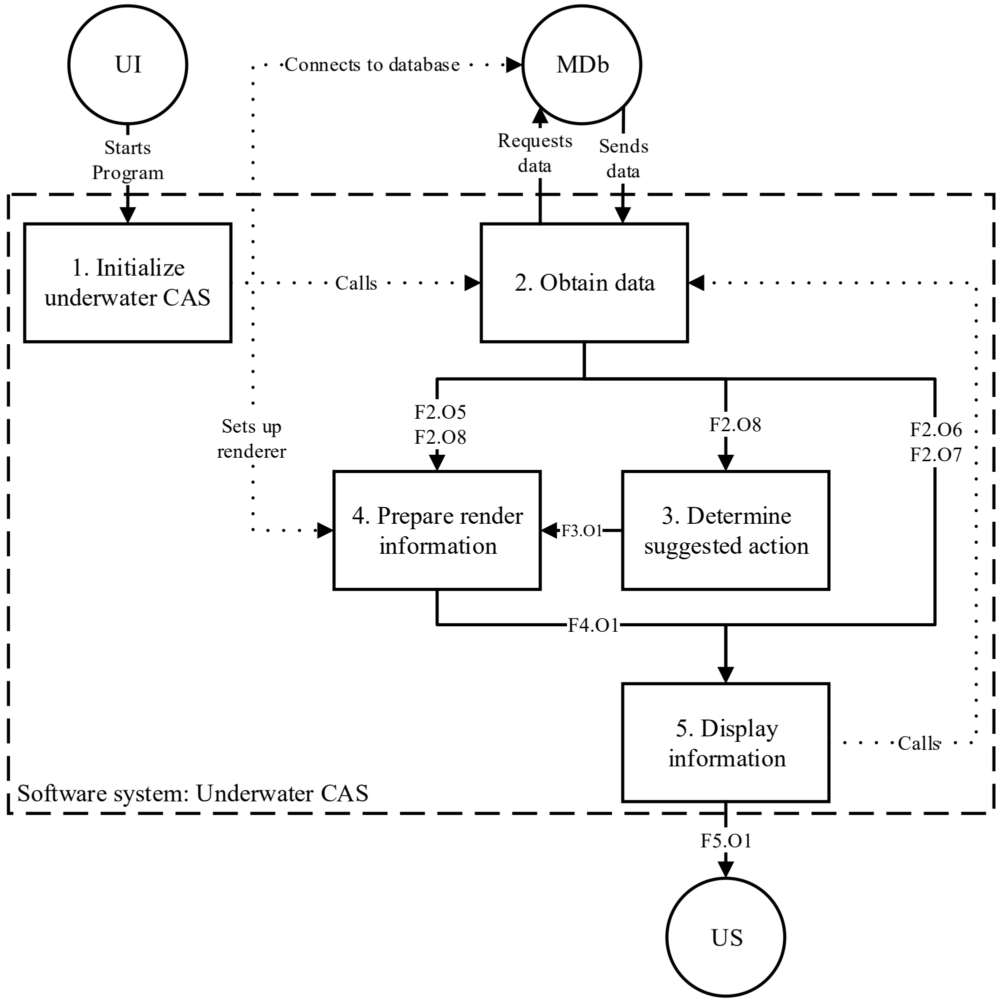


Figure 6 Functional software model for the case study underwater collision avoidance system. Reproduced from Thieme et al. (submitted-b).

In the third step (cf. Figure 4), the failure modes that apply to each software function are identified. As mentioned, there are only a few taxonomies that state the level of software analysis for which they are developed. Hence, a failure mode taxonomy is synthesised for the four types of failure modes that were investigated, which are function, interaction, time-related, and value-related failure modes.

The failure modes are derived from the literature by assessing previously presented failure modes for their applicability to the four failure mode types. The resulting list of failure modes is generic. However, some differentiations are highlighted through refined failure modes since these allow for a detailed assessment of failure modes. Refined failure modes may also imply different failure mode propagation behaviours in the next step. The full list of failure modes is presented in Article 2.

Functional software failure modes need to be identified with respect to the context of the software and the scenario. In some cases, it may be sufficient to speak of an incorrect value of an output. In another case, this might be too generic, and the incorrect value may have different implications, for example, that, above a certain value, no meaning is assigned or another action is executed.

In Step 4 of the process, the failure modes are propagated through the software systems. This is supported by the functional software model that was built in Step 2. The aim of the propagation is to identify the consequences and effects of the failure modes on the overall software system output and on the external interfaces. The software outputs and effects on the external interfaces will have different risk relevant implications, according to the context and situation. The effects on the external interfaces that are revealed with the propagation can be used further in the risk analysis process (Step 5).

The failure modes propagate through the software system according to the predefined behaviour. Wei (2006) described such failure propagation mechanisms. However, the previously described failure mode taxonomy (from Article 2) contains more and different failure modes than those for which Wei (2006) described the propagation behaviour. The propagation behaviours for the failure modes, which were not described by Wei (2006), are part of the work carried out in this thesis and are found in Article 3.

The incorporation of the identified failure events in risk analysis with methods, such as FTA or ETA, comprises Step 5 of the suggested process. All identified failure events that have been found in the previous step are reviewed, and relevant failure events for the context are included in the chosen risk model. Some iterations between the failure mode propagation (Step 4) and the risk analysis may be necessary to capture all relevant events.

The process is tested on the underwater CAS software. The results from the failure mode propagation are used to compliment a FTA to analyse how the underwater CAS may lead to a collision with an underwater subsea structure. Part of the FTA is shown in Figure 7. All events only labelled with a number below the event description are assessed through the software failure mode propagation. The FT shows how the resulting effects on the software external interfaces from the software propagation can be incorporated in the FTA, by substituting what is generally called *software failure* with specific software failure events. The analysis is not quantified. However, assigning probabilities to these events allows fully quantifying the FT. This was not part of the work carried out in the analysis.

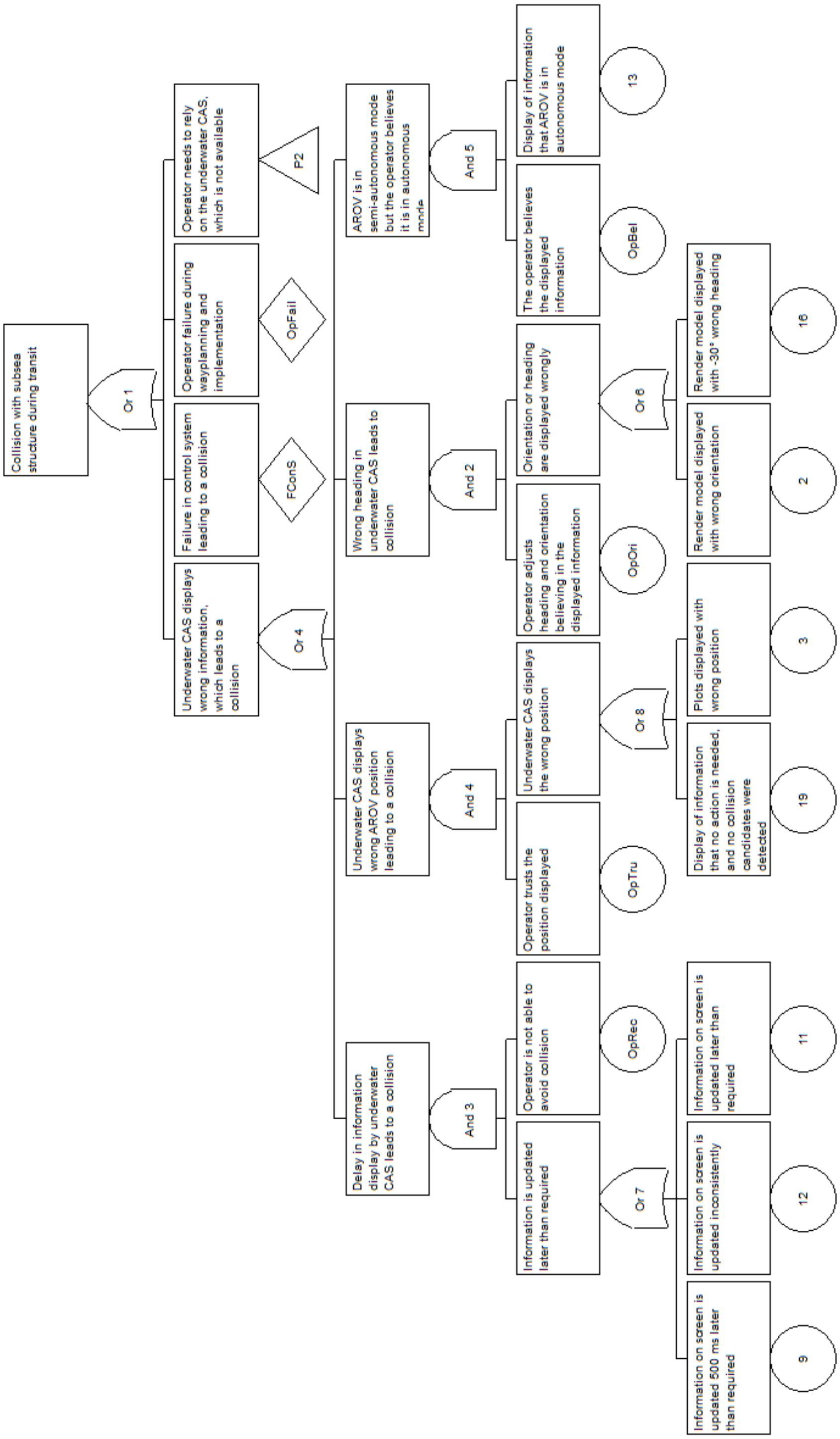

Collision with subsea structure during transit
Or 1
Operator needs to rely on the underwater CAS, which is not available
P2
Operator failure during wayplanning and implementation
OpFail
Failure in control system leading to a collision
FConS
Underwater CAS displays wrong information, which leads to a collision
Or 4
AROV is in semi-autonomous mode but the operator believes it is in autonomous mode
And 5
Display of information that AROV is in autonomous mode
13
The operator believes the displayed information
OpBel
Wrong heading in underwater CAS leads to collision
And 2
Orientation or heading are displayed wrongly
Or 6
Render model displayed with -30° wrong heading
16
Render model displayed with wrong orientation
2
Operator adjusts heading and orientation believing in the displayed information
OpOri
Underwater CAS displays wrong AROV position leading to a collision
And 4
Underwater CAS displays the wrong position
Or 8
Plots displayed with wrong position
3
Display of information that no action is needed, and no collision candidates were detected
19
Operator trusts the position displayed
OpTru
Delay in information display by underwater CAS leads to a collision
And 3
Operator is not able to avoid collision
OpRec
Information is updated later than required
Or 7
Information on screen is updated later than required
11
Information on screen is updated inconsistently
12
Information on screen is updated 500 ms later than required
9

Figure 7 Exemplary fault tree incorporating software failure events that have been derived through the software failure mode propagation for the underwater collision avoidance system software. Reproduced from Thieme et al. (submitted-b).

Based on the results of the propagation of the failure modes and the results from the risk analysis, measures for improving the software system can be identified. In most cases, these will need to address the functional failure modes by improving the software requirement specifications and software safety specifications and specifying additional functionalities that ensure safe operation of the software system. For the case study, recommendations include requirements for data validation and time outs. The last step of the suggested process is to update the analysis. This step is necessary to account for changes in the software or its context of use.

Discussion

Article 2 presents the failure mode taxonomy that was developed from the literature to address failure modes of software. So far, no clear definition of the functional level and the failure modes that apply to that level had been ventured. The definition of the functional level of software was a prerequisite for developing the failure mode taxonomy, which is suitable and unambiguous for the functional level.

One challenge in developing the failure mode taxonomy is the differentiation of failure modes from failure causes and failure effects. Through the categorisation of relevant failure mode types in function, interaction, timing-related, and value-related failure modes, and the clear definition of a software function, this was possible. Refined failure modes have been described and included in the taxonomy to highlight special cases of the failure modes. In many cases, these refined failure modes retain knowledge and implications that are relevant for specific contexts.

Another challenge with the functional view on software is the depth of analysis that one may take. There is no guideline available for how detailed a software system should be analysed. Depending on the purpose of the risk analysis, the system complexity and the available information, which correspond to the development stage of the software, the level of decomposition needs to be chosen.

The failure mode propagation for the failure modes is adopted from the literature (Wei, 2006) as much as applicable. For the failure modes that were not covered by Wei (2006), the failure mode propagation behaviour is defined. The failure mode propagation is an essential part of the overall process of including software in the risk analysis of technical systems. The propagation behaviour is described generically. However, for each case, the applicability of the description needs to be assessed to ensure that the software behaviour is reflected sufficiently.

Table 6 assesses the proposed process for including software in the risk analysis against a set of criteria that should be met by a risk analysis process including software risk. The requirements are developed from the literature (Garrett and Apostolakis, 1999; Hewett and Seker, 2005; Chu et al., 2009). The requirements aim at features and elements that are necessary to prove a sound basis for software risk assessment.

Table 6 Assessment of the proposed process for incorporating software in a risk analysis against criteria for such a process. Reproduced and adapted from Thieme et al. (submitted-b).

Requirement		Fulfilment	Comment
R1	Identify failure modes	Yes	Individual function failure modes are identified for each function. Article 2 identifies a comprehensive and coherent set of software failure modes.
R2	Identify possible failure causes	Yes	Failure causes can be found in the external interfaces, in the software itself, or in missing support. The process in these articles outlines possible failure causes.
R3	Identify consequences of failure modes	Yes	Through consistent application of the failure propagation behaviour, the consequences of software failure can be identified. These can consequently be integrated into risk models.
R4	Represent functional behaviour	Yes	The functional behaviour of the software system is explicitly modelled and represented in the functions.
R5	Represent temporal behaviour	Partly	The temporal behaviour is included in the functional software model through timing constraints and requirements and timing-related failure modes.
R6	Represent context of use	Yes	The context of use of the software is represented by including interfaces in the functional software model, considering the overall requirements, and using context-specific failure modes for certain situations.
R7	Quantify the likelihood of consequences	No	The process considers the integration of software in risk analysis. This allows for quantification of the risk model for the complete risk analysis. However, the quantification process is not covered in this article for brevity.
R8	Be modular	Yes	The functional software model is modular through the functional decomposition. Each function is represented as its own module.
R9	Be scalable		The process is scalable. It can be used for large and small software systems. The interactions between the functions are known and hence can be modelled. The analysis can focus on different levels of detail and functional decomposition.
R10	Make use of all available information	Yes	The process uses and reflects all the information that is collected in the software specifications and other documentation.
R11	Be applicable throughout the software life cycle	Yes	Through the scalability and modularity, the process can be applied at different stages of development. Especially in the operation phase, modularity makes it easy to adapt the functional software model to changes.

All requirements are fulfilled, except for R5 and R7. Requirement 7 refers to the quantification of the software failure events. This has not been executed but is assumed to be possible since software reliability and estimation methods for determining the likelihood of software flaws and errors exist and are in use.

In addition, R5 refers to the temporal behaviour of software, which is only partly met. The timing aspect of the software is incorporated through the timing-related failure modes and the associated propagation behaviour. However, dynamic risk models that are timed are needed to completely grasp the temporal aspects of the software.

The other requirements are fulfilled and addressed sufficiently. The suggested process for incorporating software failures in risk analysis provides the possibility to assess failures (through failure modes) and their consequences. Failures may be traced back to identify failure causes (R1 through R3). The process represents the functional behaviour and the context (R4 and R6) in the failure mode identification and propagation. The functional software model and the process are modular and scalable (R8 and R9). This can be attributed to the functional view.

The functional view also allows using the method in different life-cycle phases (R11), for example, during the early design, or when the software being analysed is already in use (R10). The method uses all available information, building the model and assessing failure modes based on that information. The proposed process requires a good understanding of the software and software developing process.

The failure mode propagation behaviours described may be used to identify how a failure in an external interface that gives input to the software system under analysis will affect the software output. This process is not further described and hence is not covered further.

4.3.2 Integration of Human Operator-Related Risk-Influencing Factors in Risk Models

Article 6 presents a risk management framework for UUV operation that emphasises the need to include the human-operator influence on the risk level in risk assessment. Figure 8 shows the developed risk management framework. It is based on the generic risk management framework presented in ISO 31000 (2009).

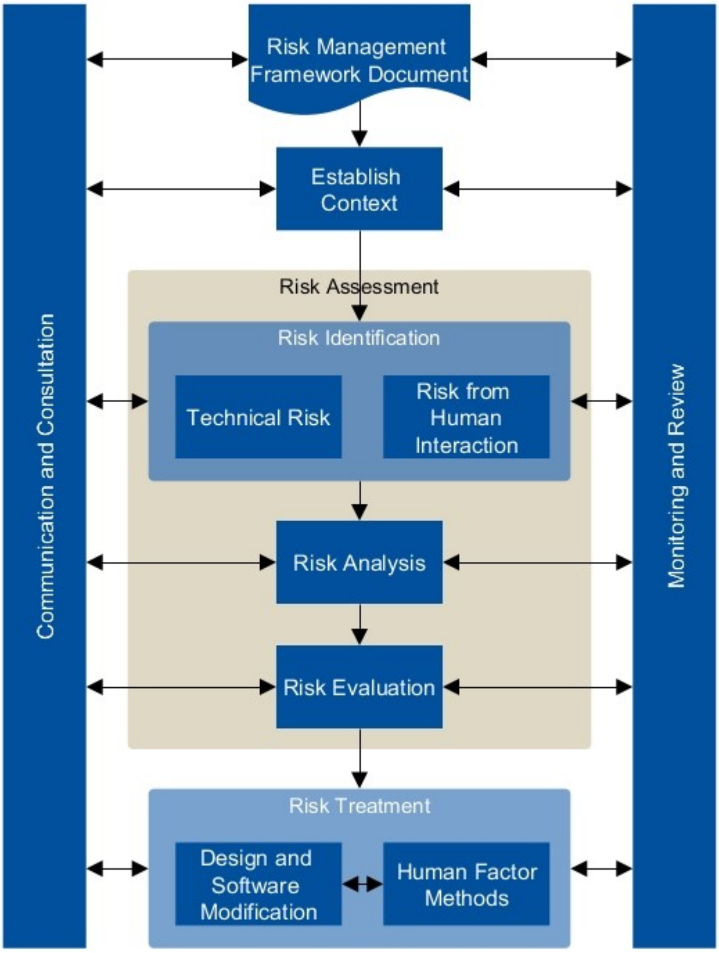


Figure 8 Risk management framework for unmanned underwater vehicle operations, highlighting the need for consideration of the risk contribution by human operators. Adapted and extended from ISO (2009). Reproduced from Thieme et al. (2015a).

Two steps are inserted that explicitly demand consideration of the human operators as contributors to the level of risk in the risk identification phase and as contributors to risk mitigation. The risk management framework document is added to the top of the risk management framework. This emphasises the need to document risk assessments and knowledge gained in relation to the risk of operation from experience, risk assessments, or external sources. This is an often neglected aspect when operating UUVs.

A case study on an AUV of the AUR Lab demonstrates the process. A preliminary hazard analysis identifies 37 hazards. The hazards with the highest level of risk are related to damage during transport, incorrect setup of the vehicle, and unexpected behaviour of the vehicle during a mission.

Based on these hazards, three accidental events are identified. These are further investigated with FTA, ETA, SPAR-H, expert judgement, and literature data. These three events are the following: an *AUV is deployed with compromised watertightness*, *AUV is deployed with the wrong setup for the target area*, and *internal faults occur in the AUV during the mission*. Each event may lead to loss of the AUV. The first two events are analysed with FTA and ETA, and the third event uses published data from Griffiths et al. (2009). For the quantification of the FTA and ETA, SPAR-H and expert judgement are used.

Table 7 summarizes the quantitative results from the risk analysis and the expected effect of risk reduction measures. Several recommendations, including updating and developing procedures for different aspects of maintenance, preparation, and operation, are issued based on the results. The results indicate that it is possible to improve UUV operation by considering human-operator-related RIFs and consequently take action to improve these RIFs and reduce the overall risk level.

Table 7 Resulting probabilities from the risk analysis of three identified accident events in Article 6 for an average autonomous underwater vehicle mission of the AUR Lab of NTNU. Reproduced and adapted from Thieme et al. (2015a).

	Consequences	***AUV is deployed with compromised watertightness***	***AUV is deployed with wrong setup for target area***	***Internal faults in the AUV during mission***
Initial assessment	Loss of AUV	1.628E-04	1.059E-03	1.600E-02
	Mission abort	2.633E-01	1.041E-01	-
	Finished mission with fault	7.979E-03	7.383E-04	-
Expected risk level through risk reduction measures	Loss of AUV	9.181E-05	8.116E-04	1.600E-02
	Mission abort	1.484E-01	7.984E-02	-
	Finished mission with fault	4.498E-03	5.099E-04	-

Article 4 presents the HAC BBN for assessing the collaborative performance between human operators and AMSs. A case study on AUV operation exemplifies the use of the BBN. The BBN contains RIFs that influence HAC. This article significantly extends the conference article by Thieme et al. (2015b).

Figure 9 shows the resulting HAC BBN. The top node is defined as *HAC performance* and has two states, namely, *adequate* and *inadequate*. The light-grey shaded nodes represent parent nodes without parents themselves, the input nodes. The white nodes are the intermediated nodes; these are influenced by the parent nodes. These nodes represent RIFs that are influenced by several other RIFs and are used to structure the influence of the individual RIFs on HAC performance.

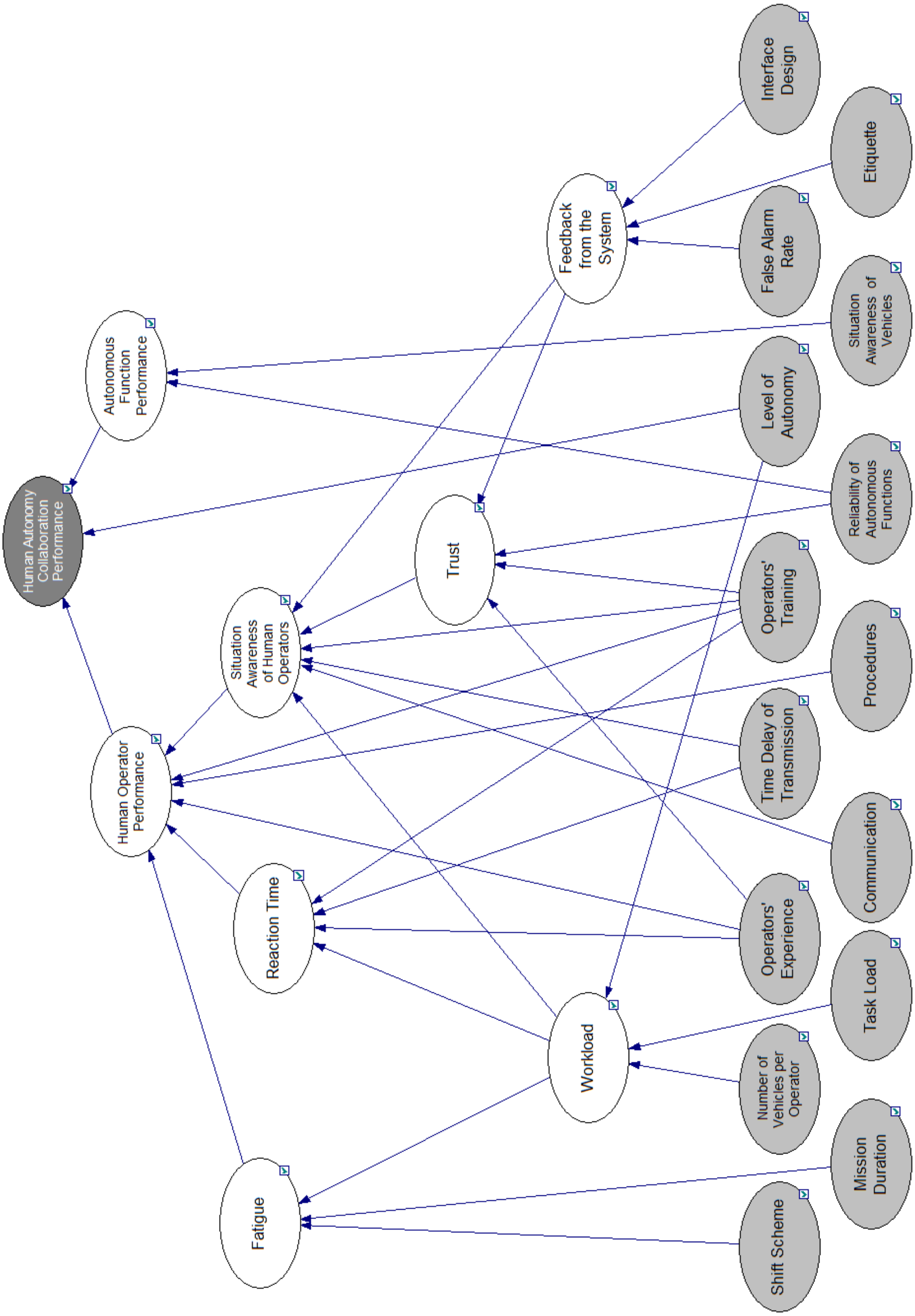

Human Autonomy Collaboration Performance
Autonomous Function Performance
Human Operator Performance
Situation Awareness of Human Operators
Reaction Time
Fatigue
Trust
Feedback from the System
Workload
Interface Design
False Alarm Rate
Level of Autonomy
Operators' Training
Time Delay of Transmission
Operators' Experience
Number of Vehicles per Operator
Shift Scheme
Etiquette
Situation Awareness of Vehicles
Reliability of Autonomous Functions
Procedures
Communication
Task Load
Mission Duration

Figure 9 Bayesian belief network representing human autonomy collaboration performance. Reproduced from Thieme and Utne (2017a).

The BBN contains 15 input nodes and eight intermediate nodes that influence HAC performance. The model introduces RIFs in the BBN that have not been considered previously for AUV operation, such as *human fatigue*, *shift scheme*, or *situational awareness of the vehicles*.

The conditional probability tables in the HAC BBN are fully quantified. The qualitative information from the literature provides the relationships and their strength. The case study shows how information on operation can be used to assess HAC performance for a specific operation. For the case study object, a REMUS 100 AUV was operated by the AUR Lab at NTNU, and HAC performance was determined to be 28.5% *inadequate* and 71.5% *adequate*.

A sensitivity analysis of the BBN reveals that the reliability of autonomous functions and autonomous capabilities of the AUV are highly influential. Regarding the RIFs related to human operators, experience and training are highly influential. Figure 10 visualizes the sensitivity analysis and presents the case study results in more detail, that is, the initial states of the nodes and states of the intermediate nodes. The more intensive red node influences HAC performance more when changing its state. When all input nodes are in the best state, HAC performance improves to 95.1% *adequate* in the case study. On the other hand, the worst states of the input nodes will lead to an *adequate* HAC performance of 23.4%.

The results show that it is possible to improve the operational performance of AUV operation through improving RIFs related to human operators. On the other hand, the technical RIFs, especially the *LoA*, the *reliability of the autonomous functions*, and *the situational awareness of the vehicles,* are major contributors to HAC performance. This is moderated through the LoA; the higher the LoA, the lower the human-operator influence on the risk level and the more important the AMS individual performance is.

Discussion

The case study in Article 6 assesses the risk with respect to loss of the AUV and mission abort, using traditional risk analysis methods. The quantification is assumed to be too conservative since the operational experience of the AUR Lab does not show as many mission aborts or losses as indicated. Only little data are available to confirm the appropriateness of the assessed values. The SPAR-H was developed for the operation of nuclear power plants. Hence, it may introduce uncertainties of unknown quantity in the probability assessment. Similarly, the expert judgement introduced uncertainty. It is based on one operator of the AUR Lab, a group assessment was not possible due to limited availability of operators of the AUR Lab.

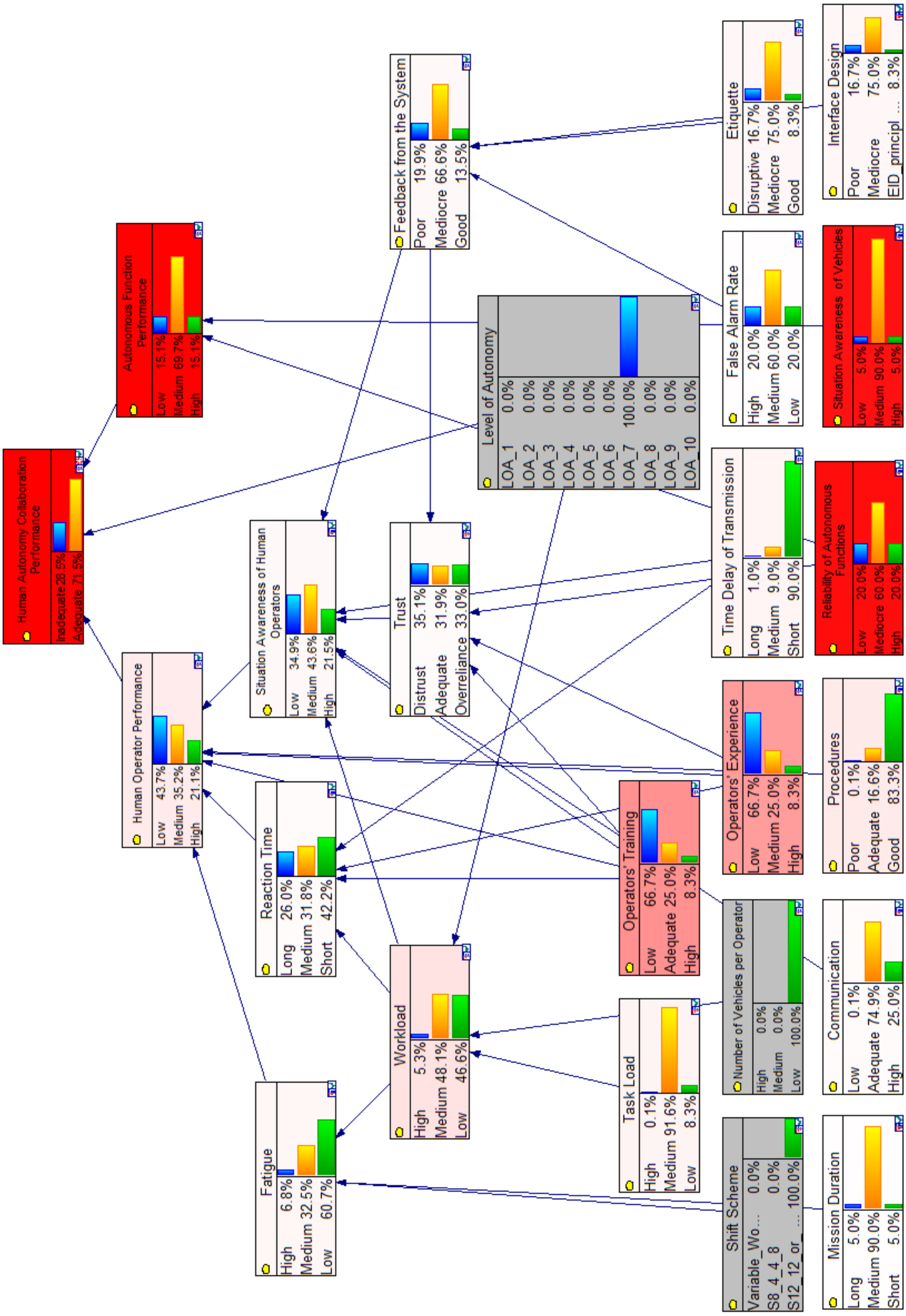


Figure 10 Resulting human autonomy collaboration Bayesian belief network for the case study. Dark red nodes have a high potential to influence human autonomy collaboration performance. The lighter the red color, the less the nodes influence human autonomy collaboration performance. Dark grey nodes have a deterministic state and cannot be varied. Reproduced from Thieme and Utne (2017a).

The HAC BBN in Article 4 focuses on AUV operation. However, the information is derived from literature on highly automated, autonomous, and remotely controlled systems from different industries. The case study demonstrates that the HAC BBN can produce meaningful results, showing that the transfer of knowledge between different autonomous systems is possible. It is assumed that the developed model can be transferred to other AMSs, such as MASSs.

The model has been validated with different qualitative tests. The tests indicate that the model is reflecting the interactions between human operators and the AUV sufficiently. The model could not be validated quantitatively since not enough data are available.

The HAC BBN does not include environmental RIFs and technical RIFs. The environment for the human operators and for the AUV itself need to be considered in a full risk analysis. In addition, technical RIFs, such as hardware failures, are not considered. This limits the assessment of the full circumstances, which might lead to a loss of an AUV. However, this work can be considered a starting point for extension.

4.4 Contribution to Research Objective 4

The fourth research objective is to develop a risk monitoring approach based on safety indicators for AMSs. This objective is addressed through Article 5. Section 2.4 summarizes the background for the work. No safety indicator approaches had been described for AMSs or MASSs.

Article 5 presents a process to identify safety indicators that can be used during the operation of an AMS to monitor the operational safety of the AMS. The process is developed through combining and integrating two existing safety indicator methods and extending their scope. The methods are the dual assurance method by the Health and Safety Executive and Chemical Industries Association (2006) and the resilience-based early warning indicator method (Øien et al., 2010; Øien and Paltrinieri, 2012). Resilience is the ability of an organisation to recognise and adapt to unexpected changes in the operational situation, to handle such changes, and to avoid an accident (Woods, 2006). The safety indicators focus on the operation of AMSs on a company and system level.

Figure 11 shows the resulting process for safety indicator development. It comprises five main steps with several sub-steps. Step 1 covers the organisational arrangements that are necessary to develop, implement, and monitor the safety indicators. In the second step, the scope of the safety indicators is defined, including the hazards that are covered by them.

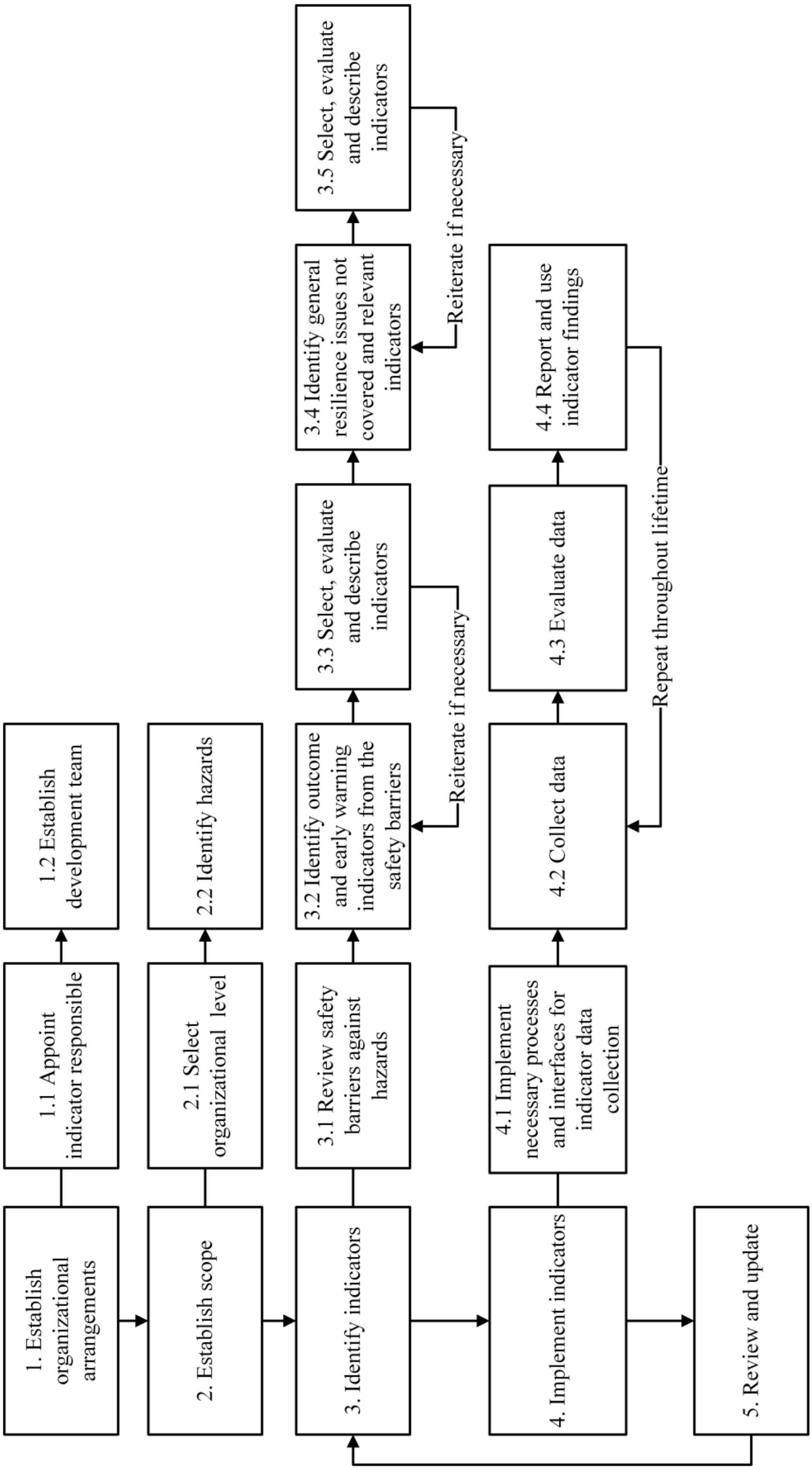

1. Establish organizational arrangements
1.1 Appoint indicator responsible
1.2 Establish development team
2. Establish scope
2.1 Select organizational level
2.2 Identify hazards
3. Identify indicators
3.1 Review safety barriers against hazards
3.2 Identify outcome and early warning indicators from the safety barriers
3.3 Select, evaluate and describe indicators
Reiterate if necessary
3.4 Identify general resilience issues not covered and relevant indicators
3.5 Select, evaluate and describe indicators
Reiterate if necessary
4. Implement indicators
4.1 Implement necessary processes and interfaces for indicator data collection
4.2 Collect data
4.3 Evaluate data
4.4 Report and use indicator findings
Repeat throughout lifetime
5. Review and update

Figure 11 Steps in the developed safety indicator identification process for autonomous marine systems. Reproduced and corrected from Thieme and Utne (2017b).

In Step 3, the indicators are identified. First, the outcome and early warning indicators are identified. When sufficient early warning indicators for each safety control function are identified, each of the early warning indicators is associated with one of the attributes of resilience. For the resilience attributes that are not associated with indicators, resilience indicators are identified (Step 3.4). This ensures that all aspects of resilience are covered, while having a manageable set of indicators.

The fourth step is the use of the indicators. Data need to be collected, and indicators need to be evaluated. Actions need to be taken if the indicators are exceeding their thresholds. The indicator system needs to be reviewed and updated regularly, assessing the usability of indicators and replacing non-useful ones. This is the last step, which might initiate another iteration of indicator identification and selection (Step 3).

A case study on a REMUS 100 AUV of the AUR Lab demonstrates the process. Sixteen indicators are identified for the AUV. The indicators address the loss of the AUV as safety-related event. Table 8 summarises the indicators and their assessment interval. There are five outcome indicators that reflect unwanted events that may happen in relation to safety. Five early warning indicators reflect safety relevant events that, if the indicator values drop below the threshold, indicate that the operation is leaving the safe operational limits. Resilience indicators are similar. However, they address the attributes of resilience that were not covered by the early warning indicators. Six resilience indicators were identified in the case study. The full case study application of the process is described in Article 5 in Part II.

Discussion

The developed process merges and adapts two existing methods for safety indicator development. The case study demonstrates that the process for developing safety indicators can be applied to identify meaningful indicators. However, not all indicators could be measured initially since not all necessary information was collected or available. In addition, several indicators are collected on a monthly basis, which makes them only limitedly suitable for mission risk monitoring in real time. For this purpose, more real-time-related indicators should be collected for future implementation of the indicator system.

The case study demonstrates that a manageable set of indicators can be developed with the process. As a manageable limit, Øien et al. (2012) suggested around 20 indicators on one activity level. The developed indicators are complementary. Important synergies that lead to better coverage of safety-related issues is achieved.

Table 8 Indicators developed for the REMUS 100 autonomous underwater vehicle of the AUR Lab at NTNU. Reproduced from Thieme and Utne (2017b).
Abbreviations: O – Outcome indicator, EW – Early Warning indicator, R – Resilience indicator.

	Outcome indicator	**Sampling interval**
O1	Number of faults that can be traced back to erroneous or lack of maintenance	Monthly
O2	Number of incidents where necessary procedures were not available during a mission	Monthly
O3	Number of times water detection sensors inside the AUV did not detect water intrusion	Monthly
O4	Percentage of missions where connection between human operators and AUV was lost (unplanned) for more than 30 minutes	Monthly
O5	Number of (temporary) losses of AUV	Monthly
EW1	Percentage of maintenance and inspections completed in specified periods	Monthly
EW2	Percentage of procedures updated and revised in the designated periods	Monthly
EW3	Percentage of time that critical sensors work without fault	During or after a mission
EW4	Percentage of anticipated status messages received from the AUV	During or after a mission
EW5	Percentage of successful recoveries of AUV within 15 minutes after the end of a mission or preliminary mission abort	Monthly
R1	Percentage of missions that have been discussed in terms of hazards and risks before mission start	Monthly
R2	Number of contacts between AUV and seafloor per hour during a mission	After a mission
R3	Percentage of missions where environmental conditions exceeded the allowable limits	Monthly
R4	Average time between status messages	During or after a mission
R5	Percentage of missions where monitoring laptop was (partly) unavailable during a mission (e.g., due to low battery)	Monthly
R6	Number of alternatively available communication channels between AUV and human operators during a mission	During or after a mission

The case study focuses on AUV operation, which is rather simple in comparison to a MASS. However, it is exhibiting similar features of operation. Similar indicators are found in the literature on performance monitoring of MASSs (Rødseth et al., 2007). Hence, it is believed that the current approach can complement such efforts.

5 Conclusion

5.1 Scientific Implications

This thesis and the associated articles contribute to the field of risk analysis and risk modelling of AMSs. This thesis identifies RIFs that need to be addressed in risk models for AMSs. Three main areas are identified that are insufficiently addressed. These are (i) the software that controls AMSs and that is used in decision-support systems for AMSs, (ii) the interaction between the human operators and the AMSs, and (iii) risk monitoring of AMSs during operation.

For including the risk contribution of software to the level of risk, a process is presented that builds on functional software failures and the propagation of these. The process of incorporating software in risk analysis is based on the software functions and associated failure modes. Software functions, with respect to risk analysis, are defined, and four types of failure modes are identified. The literature on software failure modes, except for a few exceptions, does not use a clearly defined level of assessment for the identification of failure modes. Hence, failure mode taxonomies have been reviewed and a generic set of failure modes for software that suits the functional approach is presented. To assess the effect on external interfaces, the propagation behaviour through the software functions for the software failure modes is defined for the failure modes that have not been addressed before. These effects can be incorporated into traditional risk analysis methods.

The process for incorporating software failure in risk analysis allows for assessing and including software failures from an early development stage on since it is based on the software functions and requirements and not on the implementation of the software. The process is modular and scalable, allowing for a flexible identification of failure modes, failure consequences, and failure causes. The functional behaviour of the software is completely reflected, whereas the temporal behaviour of the software is only addressed to a limited extent. The suggested process to incorporate software in risk analysis uses all information available at the time of assessment. The only criterion that was not addressed is the quantitative aspect of risk analysis for the software contribution. This should be addressed in future work.

To include the interaction between the human operators and AMSs in risk modelling, two contributions are made. First, a risk management framework is presented that highlights the need to include the RIFs regarding human operators in risk analysis of AMSs. The human operators will take a supervisory role in MASSs; hence, this contribution is also relevant for AMSs.

The second contribution to include the human-operator interactions with AMSs, is the HAC BBN, which represents the collaborative performance of human operators interacting with AMSs during operation. For the model, RIFs that affect HAC performance are identified and included in a BBN. Besides technical RIFs, pertaining to the autonomous system, such as the *reliability of the autonomous functions*, several human operator-related RIFs have been identified for the HAC BBN. These are, for example, the *human operators' experience* and *training*, *fatigue*, *task load*, and the *mission duration*. These RIFs have implications for the design of AMSs and the training of human AMS operators. The BBN is quantified for the operation of an AUV but may be adapted and further developed to other AMSs.

Having identified and modelled these RIFs, this thesis also presents an approach to develop and implement safety indicators for AMSs. These safety indicators are capable of covering the two other identified areas that need attention during risk analysis and to ensure safe operation. Different perspectives on risk are taken to ensure wide coverage of relevant RIFs. The safety indicators can be developed specifically for AMSs and assist in ensuring safe operation of the AMS.

5.2 Practical Implications for the Industry

Three main implications can be highlighted for the industry that develops AMSs, especially MASSs. Much effort is focused on the development of the algorithms and control regimes for AMSs. However, these efforts should be accompanied by assurance that the software is not contributing excessively to the level of risk. Risk analysis will be necessary for MASSs to be accepted by the national and international authorities, who will need to give permission for operation of MASSs in public waters. To accept MASSs, the public also needs to be convinced of the safety of MASSs. This thesis describes a tool to integrate the software in risk models, which allows for the identification of risk mitigation measures from an early point in system development.

Second, the implications summarised in the HAC BBN must be considered when designing AMSs, especially the HMI. The human operators are not just an addition but will take recovery actions, in case the AMS fails to handle a situation. Hence, keeping them aware of the situation and able to respond quickly to such situations is an important design consideration.

Lastly, safety monitoring of AMSs is very important. If safety indicators indicate a deterioration of operational safety performance, measures should be taken before accidents happen. Safety indicators are also an important tool to keep the human operators alert and aware of the situation.

Designers, owners, and operators of MASSs and other AMSs may use the findings in this thesis to include them in risk assessments and analyses. The implementation of the findings will contribute to safe designs and safe operations. The findings may also be included in software tools for risk assessment, or risk monitoring.

5.3 Further Work

This thesis presents models and processes to assess and analyse the risk of AMSs, especially AUVs and MASSs. The AUVs exist already and are used, while MASSs are still conceptual. Hence, further work with respect to risk analysis of MASSs needs to gather necessary data and develop methods and models further, such that the development of safe MASSs can be assured. The cooperation between industry and academia and the involvement of important stakeholders, such as human operators, from an early development stage of AMSs are necessary to ensure the safe operation of MASSs and AMSs in general.

The process for incorporating software in risk analysis is time-consuming and the traceability for larger software systems might be tedious. Hence, a software tool should be developed that includes the failure mode taxonomy, propagation behaviour, and building blocks for the functional software model. This software tool could assist in building the functional software model, assessing possible failure modes, and supporting analysts in propagating failure modes through the software system to identify the effect on the external interfaces.

Another feature that may be included in such a software tool should analyse the effect of the propagated failure modes on the external interfaces of the software and how these failures will affect the software output. In addition, the proposed process does not allow for the quantification of software failure probabilities or frequencies yet. Hence, further research is necessary to develop a suitable quantification approach for the software failures. This will allow the analysts to include the results in quantified risk models for AMSs.

A model for HAC performance was developed. However, the mission outcome is also dependent on other kinds of RIFs. A holistic risk model should be developed including all technical RIFs related to software and hardware, human RIFs, organisational RIFs, and environmental RIFs. Only in this way may hazardous interactions be revealed and mitigating measures be identified.

The environmental aspects acting on human operators and AMSs have not been included. These interactions may be interesting for further research. A holistic model may provide more insight in the interactions between human operators, autonomous systems, and the operational environment. Some research is necessary to assess the applicability to MASSs. The HAC model includes RIFs that apply to AUVs. However, adaptions to MASSs may be necessary.

In general, more research is necessary to identify and apply suitable methods for risk assessment, risk analysis, and risk monitoring of AMSs. These methods need to be supplemented by the use of adequate testing, verification, and validation methods to ensure safe AMSs.

References

AAWA 2016. Remote and Autonomous Ships - the Next Steps. *In*: Laurinen, M. (ed.) *Advanced Autonomous Waterborne Applications.* London, UK pp. 56-73.

Abdulkhaleq, A. & Wagner, S. 2015. Integrated Safety Analysis Using Systems-Theoretic Process Analysis and Software Model Checking. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* pp. 121-134.

Abdulkhaleq, A., Wagner, S. & Leveson, N. 2015. A Comprehensive Safety Engineering Approach for Software-Intensive Systems Based on STPA. Procedia Engineering 04.-06.10.2015. pp. 2-11.

Ait Allal, A., Mansouri, K., Youssfi, M. & Qbadou, M. 2017. Toward Reliable Maritime Communication for a Safe Operation of Autonomous Ship. Ubiquitous Networking. Third International Symposium, UNet 2017, 9.-12.05.2017 Cham, Switzerland. Springer International Publishing, pp. 261-274.

Al-Dabbagh, A. W. 2009. *Dynamic Flowgraph Methodology for Reliability Modelling of Networked Control Systems.* M.A.Sc. Thesis, University of Ontario Institute of Technology, Oshawa, ON, Canada.

Al-Dabbagh, A. W. & Lu, L. 2010. Reliability Modeling of Networked Control Systems Using Dynamic Flowgraph Methodology. *Reliability Engineering & System Safety,* 95, pp. 1202-1209.

Aldemir, T., Guarro, S. B., Kirschenbaum, J., Mandellil, D., Mangan, L. A., Bucci, P., Yau, M. K., Johnson, B., Elks, C., Ekici, E., Stovsky, M. P., Miller, D. W., Sun, X., Arndt, S. A., Nguyen, Q. & Dion, J. 2009. A Benchmark Implementation of Two Dynamic Methodologies for the Reliability Modeling of Digital Instrumentation and Control Systems. Washington, DC, USA: Office of Nuclear Regulatory Research. pp.

Aldemir, T., Guarro, S. B., Mandelli, D., Kirschenbaum, J., Mangan, L. A., Bucci, P., Yau, M. K., Ekici, E., Miller, D. W., Sun, X. & Arndt, S. A. 2010. Probabilistic Risk Assessment Modeling of Digital Instrumentation and Control Systems Using Two Dynamic Methodologies. *Reliability Engineering & System Safety,* 95, pp. 1011-1039.

Allianz Global Corporate & Speciality 2018. Safety and Shipping - Review 2017. *In*: Dobie, G. (ed.). Munich, Germany: Allianz Global Corporate & Specialty SE. pp. 2-39.

Authen, S., Björkman, K., Holmberg, J.-E. & Larsson, J. 2010. Guidelines for Reliability Analysis of Digital Systems in PSA Context — Phase 1 Status Report. Roskilde, Denmark: Nordik Nuclear Safety Research (NKS). pp. 1-23.

Authen, S. & Holmberg, J.-E. 2012. Reliability Analysis of Digital Systems in a Probabilistic Risk Analysis for Nuclear Power Plants. *Nuclear Engineering and Technology,* 44, pp. 471-482.

Authen, S. & Holmberg, J.-E. 2013. Guidelines for Reliability Analysis of Digital Systems in PSA Context — Phase 3 Status Report. Roskilde Denmark: Nordic nuclear safety research (NKS). pp. 3-37.

Aven, T. 2012. The Risk Concept—Historical and Recent Development Trends. *Reliability Engineering & System Safety,* 99, pp. 33-44.

Bainbridge, L. 1983. Ironies of Automation. *Automatica,* 19, pp. 775-779.

Bertram, V. 2008. Unmanned Surface Vehicles–a Survey. *Skibsteknisk Selskab, Copenhagen, Denmark*, pp. 1-14.

Bertram, V. 2016. Autonomous Ship Technology -Smart for Sure, Unmanned Maybe. *In*: Griffiths, M. (ed.) *Smart Ship Technology.* London, UK: The Royal Institute of Naval Architects. pp. 5-112.

Brito, M. P. & Griffiths, G. 2009. Results of Expert Judgments on the Faults and Risks with Autosub3 and an Analysis of Its Campaign to Pine Island Bay, Antarctica, 2009. Proceedings of the

International Symposium on Unmanned Untethered Submersible Technology (UUST 2009),, 23.-26.08.2009 Durham, New Hampshire. Autonomous Undersea Systems Institute (AUSI), pp. 1-14.

Brito, M. P. & Griffiths, G. 2016a. Autonomy: Risk Assessment. *In:* Dhanak, M. R. & Xiros, N. I. (eds.) *Springer Handbook of Ocean Engineering.* Berlin Heidelberg: Springer International Publishing. pp. 527-544

Brito, M. P. & Griffiths, G. 2016b. A Bayesian Approach for Predicting Risk of Autonomous Underwater Vehicle Loss During Their Missions. *Reliability Engineering & System Safety,* 146, pp. 55-67.

Brito, M. P. & Griffiths, G. 2018. Updating Autonomous Underwater Vehicle Risk Based on the Effectiveness of Failure Prevention and Correction. *Journal of Atmospheric and Oceanic Technology,* 35, pp. 797-808.

Brito, M. P., Griffiths, G. & Challenor, P. 2010. Risk Analysis for Autonomous Underwater Vehicle Operations in Extreme Environments. *Risk Analysis,* 30, pp. 1771-1788.

Bureau Veritas 2017. Guidelines for Autonomous Shipping. Paris, France. pp. 5-25.

Chellali, R. & Baizid, K. 2011. What Maps and What Displays for Remote Situation Awareness and ROV Localization? *In:* Salvendy, G. & Smith, M. (eds.) *Human Interface and the Management of Information. Interacting with Information.* Springer Berlin Heidelberg. pp. 364-372

Chen, J. Y. C., Barnes, M. J. & Harper-Sciarini, M. 2011. Supervisory Control of Multiple Robots: Human-Performance Issues and User-Interface Design. *IEEE Transactions on Systems, Man and Cybernetics Part C: Applications and Reviews,* 41, pp. 435-454.

Chen, J. Y. C., Haas, E. C. & Barnes, M. J. 2007. Human Performance Issues and User Interface Design for Teleoperated Robots. *IEEE Transactions on Systems, Man and Cybernetics Part C: Applications and Reviews,* 37, pp. 1231-1245.

Christ, R. D. & Wernli, R. L. 2007. *The ROV Manual,* Oxford, Butterworth-Heinemann.

Chu, T.-L., Martinez-Guridi, G., Yue, M., Samanta, P., Vinod, G. & Lehner, J. 2009. Workshop on Philosophical Basis for Incorporating Software Failures in a Probabilistic Risk Assessment. *Digital System Software PRA.* Brookhaven National Laboratory. pp. 1-1-2-21.

Chu, T.-L., Yue, M., Martinez-Guridi, G. & Lehner, J. 2010. Review of Quantitative Software Reliability Methods. U.S. Nuclear Regulatory Commission, Office of Nuclear Regulatory Research, Division of Risk Analysis. pp. 1-98.

Creswell, J. W. 2014. *Research Design : Qualitative, Quantitative, and Mixed Methods Approaches,* London, UK; New Delhi, India; Singapore, Sage Publications Inc.

Cummings, M. 2014. Man Versus Machine or Man Plus Machine? *Ieee Intelligent Systems,* 29, pp. 62-69.

Danish Maritime Authority 2018. Analysis of Regulatory Barriers to the Use of Autonomous Ships. *Regulatory scoping exercise for the use of maritime autonomous surface ships (MASS).* Danmark: Maritime Safety Committee, Danish Maritime Authority (DMA). pp. 1-93.

DNV-Gl. 2015. *The Revolt* [Online]. Det Norske Veritas and Germanischer Lloyd. Available: https://www.dnvgl.com/technology-innovation/revolt/index.html [Accessed: 23.07.2015].

DNV-Gl. 2018. *Unmanned Ships on the Horizon* [Online]. Det Norske Veritas and Germanischer Lloyd. Available: https://www.dnvgl.com/article/unmanned-ships-on-the-horizon-94273 [Accessed: 24.04.2018].

Earthy, J. V. & Lützhöft, M. 2018. Autonomous Ships, Ict and Safety Management. *In:* Oltedal, H. A. & Lützhöft, M. (eds.) *Managing Maritime Safety.* Oxon, UK; New York, NY, USA: Routledge. pp. 143-165

EN 2004. NS-En14514: Space Engineering Standards - Functional Analysis. Brussels, Belgium: European Committee for Standardization. pp.

Endsley, M. R. & Kaber, D. B. 1999. Level of Automation Effects on Performance, Situation Awareness and Workload in a Dynamic Control Task. *Ergonomics,* 42, pp. 462-492.

European Commission. 2018. *Reducing Emissions from the Shipping Sector* [Online]. European Commission. Available: https://ec.europa.eu/clima/policies/transport/shipping_en [Accessed: 30.04.2018].

Gander, P., Hartley, L., Powell, D., Cabon, P., Hitchcock, E., Mills, A. & Popkin, S. 2011. Fatigue Risk Management: Organizational Factors at the Regulatory and Industry/Company Level. *Accident Analysis & Prevention,* 43, pp. 573-590.

Garrett, C. J. & Apostolakis, G. 1999. Context in the Risk Assessment of Digital Systems. *Risk Analysis,* 19, pp. 23-32.

Garrett, C. J., Guarro, S. B. & Apostolakis, G. E. 1995. The Dynamic Flowgraph Methodology for Assessing the Dependability of Embedded Software Systems. *IEEE Transactions on Systems, Man, and Cybernetics,* 25, pp. 824-840.

Gertman, D., Blackman, H., Marble, J., Byers, J. & Smith, C. 2005. Nureg/Cr-68832005: The SPAR-H Human-Reliability Analysis Method. Washington, DC: NUREG, U.S.NRC. pp. 5-64.

Griffiths, G., Brito, M., Robbins, I. & Moline, M. 2009. Reliability of Two Remus-100 AUVs Based on Fault Log Analysis and Elicited Expert Judgment. *Proceedings of the International Symposium on Unmanned Untethered Submersible Technology (Uust 2009), Durham, New Hampshire, 23-26 August 2009.* Durham NH, USA: Autonomous Undersea Systems Institute (AUSI). pp. [12p]

Griffiths, G. & Brito, M. P. 2008. Predicting Risk in Missions under Sea Ice with Autonomous Underwater Vehicles. *Autonomous Underwater Vehicles (AUV) 2008.* pp. 1-7.

Guarro, S. B., Yau, M. K. & Dixon, S. 2013. Context-Based Software Risk Model (Csrm) Application Guide. 1st Ed. ed. Washington, D.C. 20546: ASCA Inc. pp. 1-73.

Guarro, S. B., Yau, M. K. & Motamed, M. 1996. Development of Tools for Safety Analysis of Control Software in Advanced Reactors. Washington DC: ASCA Inc. pp. 1-115.

Harris, C. A., Phillips, A. B., Dopico-Gonzalez, C. & Brito, M. P. 2016. Risk and Reliability Modelling for Multi-Vehicle Marine Domains. *Ieee/Oes Autonomous Underwater Vehicles (Auv)*, pp. 286-293.

Hassan, J. & Khan, F. I. 2012. Risk-Based Asset Integrity Indicators. *Journal of Loss Prevention in the Process Industries,* 25, pp. 544-554.

Health and Safety Executive & Chemical Industries Association 2006. *Developing Process Safety Indicators: A Step-by-Step Guide for Chemical and Major Hazard Industries,* Norwich, Chemical Industries Association (CIA) and Health and Safety Executive (HSE).

Hegde, J. 2018. *Tools and Methods to Manage Risk in Autonomous Subsea Inspection, Maintenance and Repair Operations.* PhD Thesis, Norwegian University of Science and Technology (NTNU), Trondheim, Norway.

Hegde, J., Henriksen, E. H., Utne, I. B. & Schjølberg, I. submitted. Development of Safety Envelopes and Subsea Traffic Rules for Autonomous Remotely Operated Vehicles. *Submitted to: Journal of Loss Prevention in the Process Industries*, pp. 133-155.

Hegde, J., Utne, I. B. & Schjølberg, I. 2015. Applicability of Current Remotely Operated Vehicle Standards and Guidelines to Autonomous Subsea Imr Operations. OMAE2015, St. Johns, NF, Canada. ASME, pp. 1-10.

Hegde, J., Utne, I. B. & Schjølberg, I. 2016. Development of Collision Risk Indicators for Autonomous Subsea Inspection Maintenance and Repair. *Journal of Loss Prevention in the Process Industries,* 44, pp. 440-452.

Hegde, J., Utne, I. B., Schjølberg, I. & Thorkildsen, B. 2018. A Bayesian Approach to Risk Modeling of Autonomous Subsea Intervention Operations. *Reliability Engineering & System Safety,* 175, pp. 142-159.

Hewett, R. & Seker, R. 2005. A Risk Assessment Model of Embedded Software Systems. 2005 29th Annual IEEE/NASA Software Engineering Workshop, SEW'05, 06.-07.04.2005 Greenbelt, MD, USA. Institute of Electrical and Electronics Engineers Computer Society, pp. 142-149.

Holmberg, J.-E., Authen, S. & Amri, A. 2012. Development of Best Practice Guidelines on Failure Modes Taxonomy for Reliability Assessment of Digital Ic Systems for PSA. 11th International

Probabilistic Safety Assessment and Management Conference and the Annual European Safety and Reliability Conference, PSAM11 ESREL 2012, 25.-29.06.2012 Helsinki, Finland. Probablistic Safety Assessment and Management (IAPSAM), pp. 1887-1894.

Huang, B., Zhang, H. & Lu, M. 2009. Software FMEA Approach Based on Failure Modes Database. 8th International Conference on Reliability, Maintainability and Safety, 20.-24.07.2009. pp. 749-753.

Huang, H.-M. 2007. Autonomy Levels for Unmanned Systems (Alfus) Framework. *PerMIS'07.* Washington, D.C. USA. pp. 48-53.

IAEA 2000. Operational Safety Performance Indicators for Nuclear Power Plants *IAEA-TECDOC.* Vienna: International Atomic Energy Agency. pp. 1-75.

IEC 2010. IEC 61508: Functional Safety of Electrical/Electronic/ Programmable Electronic Safety Related Systems. Geneva, Switzerland: International Electrotechnical Commission. pp.

IEC EN 2006. EN IEC 60812: Analysis Techniques for System Reliability – Procedure for Failure Mode and Effects Analysis (FMEA). Brussels, Belgium: International Electrotechnical Commission, European Committee for Electrotechnical Standardization. pp.

IEEE 2016. IEEE Std 1633-2016: IEEE Recommended Practice on Software Reliability. New York, NY, USA: Institute of Electrical and Electronics Engineers Reliability Society. pp. 1-261.

Insaurralde, C. C. & Lane, D. M. 2012. Autonomy-Assessment Criteria for Underwater Vehicles. Autonomous Underwater Vehicles (AUV) 2012 IEEE/EOS. pp. 1-8.

International Committee of Medical Journal Editors. 2018. *Defining the Role of Authors and Contributors* [Online]. Vancouver, Canada: International Comittee of Medical Journal Editors. Available: http://www.icmje.org/recommendations/browse/roles-and-responsibilities/defining-the-role-of-authors-and-contributors.html [Accessed: 04.04.2018].

ISO 2009. ISO 31000 Risk Management - Principles and Guidelines. Geneva, Switzerland: International Organization for Standardization pp.

Jensen, F. 2015. *Hazard and Risk Assessment of Unmanned Dry Bulk Carriers on the High Seas.* M.Sc. Thesis, Technische Universität Hamburg Harburg, Hamburg, Germany.

Kaber, D. B. 2017. A Conceptual Framework of Autonomous and Automated Agents. *Theoretical Issues in Ergonomics Science*, pp. 1-25.

Kaplan, S. & Garrick, B. J. 1981. On the Quantitative Definition of Risk. *Risk Analysis,* 1, pp. 11-27.

Karwowski, W. 2006. The Discipline of Ergonomics and Human Factors. *In:* Salvendy, G. (ed.) *Handbook of Human Factors and Ergonomics.* John Wiley & Sons, Inc. pp. 1-31

Knegtering, B. & Pasman, H. 2013. The Safety Barometer. *Journal of Loss Prevention in the Process Industries,* 26, pp. 821-829.

Kongsberg Maritime. 2016. *Automated Ships Ltd and Kongsberg to Build First Unmanned and Fully Autonomous Ship for Offshore Operations* [Online]. Kongsberg Maritime. Available: https://www.km.kongsberg.com/ks/web/nokbg0238.nsf/AllWeb/65865972888D25FAC125805E00281D50?OpenDocument [Accessed: 24.04.2018].

Kongsberg Maritime. 2017. *Yara and Kongsberg Enter into Partnership to Build World's First Autonomous and Zero Emissions Ship* [Online]. online article: Kongsberg. Available: https://www.km.kongsberg.com/ks/web/nokbg0238.nsf/AllWeb/98A8C576AEFC85AFC125811A0037F6C4?OpenDocument [Accessed: 27.07.2017].

Kothari, C. R. 2004. *Research Methodology - Methods and Techniques,* New Delhi, Bangalore, Chennai, Cochin, Guwahati, Hyderabad, Jalandhar, Kolkata, Lucknow, Mumbai, Ranchi, New Age International.

Kretschmann, L., Rødseth, Ø. J., Sage Fuller, B., Noble, H., Horahan, J. & Mcdowell, H. 2015. D9.3: Quantitative Assessment. *Maritime Unmanned Navigation through Intelligence in Networks.* 1st Ed. ed. pp. 12-20.

Kretschmann, L., Rødseth, Ø. J., Tjora, Å., Sage Fuller, B., Noble, H. & Horahan, J. 2015. D9.2: Qualitative Assessment. *Maritime Unmanned Navigation through Intelligence in Networks.* 1st Ed. ed. pp. 8-12.

Lee, J. D. & See, K. A. 2004. Trust in Automation: Designing for Appropriate Reliance. *Human Factors: The Journal of the Human Factors and Ergonomics Society,* 46, pp. 50-80.

Leveson, N. G. 2004. A New Accident Model for Engineering Safer Systems. *Safety Science,* 42, pp. 237-270.

Leveson, N. G., Fleming, C. H., Spencer, M., Thomas, J. & Wilkinson, C. 2012. Safety Assessment of Complex, Software-Intensive Systems. *SAE International Journal of Aerospace,* 5, pp. 233-244.

Li, B. 2004. *Integrating Software into PRA (Probabilistic Risk Assessment).* PhD Thesis Monograph, University of Maryland, College Park, Md.

Li, B., Li, M., Ghose, S. & Smidts, C. 2003. Integrating Software into PRA. Issre 2003: 14th International Symposium on Software Reliability Engineering, Proceedings. pp. 457-467.

Li, B., Li, M. & Smidts, C. 2005. Integrating Software into PRA: A Test-Based Approach. *Risk Analysis,* 25, pp. 1061-1077.

Li, Z., Bachmayer, R. & Vardy, A. 2016. Risk Analysis of an Autonomous Surface Craft for Operation in Harsh Ocean Environments. 2016 Autonomous Underwater Vehicles (AUV), 06.-09.11.2016 Tokyo, Japan. Institute of Electrical and Electronics Engineers Inc., pp. 294-300.

Lloyd's Register 2016. Cyber-Enabled Ships - Shipright Procedure – Autonomous Ships. 1st Ed. ed. Southampton, UK: Lloyd's Register Group Ltd. pp. 1-19.

Manley, J. E. 2007. The Role of Risk in AUV Development and Deployment. OCEANS 2007 - Europe. pp. 1-6.

Manley, J. E. 2008. Unmanned Surface Vehicles, 15 Years of Development. Oceans 2008, Quebec City, CANADA. IEEE, pp. 1707-1710.

Mosleh, A. 2014. PRA: A Perspective on Strengths, Current Limitations, and Possible Improvements. *Nuclear Engineering and Technology,* 46, pp. 1-10.

MUNIN. 2012. *Maritime Unmanned Navigation through Intelligence in Networks* [Online]. Available: http://www.unmanned-ship.org/munin/ [Accessed: 23.07.2016].

Nautilus Federation 2018. Report of a Survey on What Maritime Professionals Think About Autonomous Shipping. *Regulatory scoping exercise for the use of maritime autonomous surface ships (MASS).* London, UK: International Maritime Organization, Maritime Safety Committee. pp. 4-17.

Norwegian Maritime Authority. 2016. *World's First Test Area for Autonomous Ships Opened* [Online]. Haugesund, Norway: Sjøfartsdirektoratet. Available: https://www.sjofartsdir.no/en/news/news-from-the-nma/worlds-first-test-area-for-autonomous-ships-opened/ [Accessed 07.10.2016].

OECD 2014. Failure Modes Taxonomy for Reliability Assessment of Digital I&C Systems for PRA. Paris, France: Organisation for Economic Co-operation and Development, Nuclear Energy Agency. pp. 1-135.

Ozarin, N. W. 2003. Developing Rules for Failure Modes and Effects Analysis of Computer Software. SAE Advances in Aviation Safety Conference - 2003 Aerospace Congress and Exhibition, 08.-11.09.2003 Montreal, QC, Canada. SAE International, pp.

Parasuraman, R. & Miller, C. A. 2004. Trust and Etiquette in High-Criticality Automated Systems. *Communications of the Acm,* 47, pp. 51-55.

Parasuraman, R. & Riley, V. 1997. Humans and Automation: Use, Misuse, Disuse, Abuse. *Human Factors,* 39, pp. 230-253.

Parasuraman, R. & Wickens, C. D. 2008. Humans: Still Vital after All These Years of Automation. *Human Factors: The Journal of the Human Factors and Ergonomics Society,* 50, pp. 511-520.

Park, G. Y., Kim, D. H. & Lee, D. Y. 2014. Software FMEA Analysis for Safety-Related Application Software. *Annals of Nuclear Energy,* 70, pp. 96-102.

Pedersen, P. T. 2010. Review and Application of Ship Collision and Grounding Analysis Procedures. *Marine Structures,* 23, pp. 241-262.

Prasanna, K. N., Gokhale, S. A., Agarwal, R., Chetwani, R. R., Ravindra, M. & Bharadwaj, K. M. 2014. Application of Software Failure Mode and Effect Analysis for on-Board Software. 2014

International Conference on Advances in Computing, Communications and Informatics (ICACCI), 24.-27.09.2014. pp. 684-688.

Rausand, M. 2011. *Risk Assessment - Theory, Methods, and Applications,* Hoboken, New Jersey, USA, John Wiley & Sons.

Ristord, L. & Esmenjaud, C. 2002. FMEA Performed on the Spinline3 Operational System Software as Part of the Tihange 1 Nis Refurbishment Safety Case. *Cnra/Csni Workshop on Licensing and Operating Experience of Computer-Based I&C Systems.* Hluboka nad Vltavou, Czech Republic: NEA/CSN/OECD. pp. 37-50.

Rokseth, B., Utne, I. B. & Vinnem, J. E. 2016. A Systems Approach to Risk Analysis of Maritime Operations. *Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability*, pp. 53-68.

Rokseth, B., Utne, I. B. & Vinnem, J. E. 2017. Deriving Verification Objectives and Scenarios for Maritime Systems Using the Systems-Theoretic Process Analysis. *Reliability Engineering & System Safety,* 169, pp. 18-31.

Roll-Hansen, N. 2009. Why the Distinction between Basic (Theoretical) and Applied (Practical) Research Is Important in the Politics of Science. *In*: Fennell, D. (ed.). London: Contingency And Dissent in Science Project, London School of Economics and Political Science. pp. 2-27.

Rødseth, Ø. J. & Burmeister, H.-C. 2015. Risk Assessment for an Unmanned Merchant Ship. *TransNav, the International Journal on Marine Navigation and Safety of Sea Transportation,* 9, pp. 357-364.

Rødseth, Ø. J. & Nordahl, H. 2017. Definitions for Autonomous Merchant Ships. Trondheim, Norway: NFAS - Norwegian Forum for Autonomous Ships. pp. 1-21.

Rødseth, Ø. J., Steinebach, C. & Mo, B. 2007. The Use of Technical Condition Indices in Ship Maintenance Planning and the Monitoring of the Ship's Safety Condition. International Symposium on Maritime Safety, Security and Environmental Protection, 20.-21.09.2007 Athens, Greece. pp. 1-10.

Rødseth, Ø. J. & Tjora, Å. 2014. A Risk Based Approach to the Design of Unmanned Ship Control Systems. *Maritime-Port Technology and Development.* CRC Press. pp. 153-161

Rødseth, Ø. J., Tjora, Å. & Baltzersen, P. 2014. D4.5 Architecture Specification. *Maritime Unmanned Navigation through Intelligence in Networks.* pp. 1-41.

Sheridan, T. B. 1982. Supervisory Control: Problems, Theory and Experiment for Application to Human-Computer Interaction in Undersea Remote Systems. DTIC Document. pp.

Sheridan, T. B. 2006. Supervisory Control. *In:* Salvendy, G. (ed.) *Handbook of Human Factors and Ergonomics.* 3rd ed.: John Wiley & Sons, Inc. pp. 1025-1052

Sheridan, T. B. & Parasuraman, R. 2005. Human-Automation Interaction. *Reviews of Human Factors and Ergonomics,* 1, pp. 89-129.

Sheridan, T. B. & Verplank, W. L. 1978. Human and Computer Control of Undersea Teleoperators. DTIC Document. pp.

Sklet, S. 2006. Safety Barriers: Definition, Classification, and Performance. *Journal of Loss Prevention in the Process Industries,* 19, pp. 494-506.

Skogdalen, J. E., Utne, I. B. & Vinnem, J. E. 2011. Developing Safety Indicators for Preventing Offshore Oil and Gas Deepwater Drilling Blowouts. *Safety Science,* 49, pp. 1187-1199.

Stadler, J. J. & Seidl, N. J. 2013. Software Failure Modes and Effects Analysis. 59th Annual Reliability and Maintainability Symposium, RAMS 2013, 28.-31.01.2013 Orlando, FL, United States. Institute of Electrical and Electronics Engineers Inc., pp. 1-5.

Stamatelatos, M., Dezfuli, H., Apostolakis, G., Everline, C. J., Guarro, S. B., Mathias, D., Mosleh, A., Paulos, T., Riha, D., Smith, C., Vesely, W. E. & Youngblood, R. 2011. Probabilistic Risk Assessment Procedures Guide for NASA Managers and Practitioners. Washington D.C.: National Aeronautics and Space Administration. pp. 1-431.

Stokey, R., Austin, T., Von Alt, C., Purcell, M., Goldsborough, R., Forrester, N. & Allen, B. 1999. AUV Bloopers or Why Murphy Must Have Been an Optimist: A Practical Look at Achieving Mission

Level Reliability in an Autonomous Underwater Vehicle. Proceedings of the International Symposium on Unmanned Untethered Submersible Technology, August New Hampshire. pp. 32-40.

Swuste, P., Theunissen, J., Schmitz, P., Reniers, G. & Blokland, P. 2016. Process Safety Indicators, a Review of Literature. *Journal of Loss Prevention in the Process Industries,* 40, pp. 162-173.

Thieme, C. A., Mosleh, A., Utne, I. B. & Hegde, J. submitted-a. Incorporating Software Failure in Risk Analysis – Part 1: Software Functional Failure Mode Classification. *Submitted for review to Reliability Engineering and System Safety*, pp. 1-29.

Thieme, C. A., Mosleh, A., Utne, I. B. & Hegde, J. submitted-b. Incorporating Software Failure in Risk Analysis – Part 2: Risk Modeling Process and Case Study. *Submitted for review to Reliabilty Engineering and System Safety*, pp. 1-33.

Thieme, C. A. & Utne, I. B. 2017a. A Risk Model for Autonomous Marine Systems and Operation Focusing on Human–Autonomy Collaboration. *Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability,* 231, pp. 446-464.

Thieme, C. A. & Utne, I. B. 2017b. Safety Performance Monitoring of Autonomous Marine Systems. *Reliability Engineering & System Safety,* 159, pp. 264-275.

Thieme, C. A., Utne, I. B. & Haugen, S. 2018. Assessing Ship Risk Model Applicability to Marine Autonomous Surface Ships. *Ocean Engineering,* 165, pp. 140 - 154.

Thieme, C. A., Utne, I. B. & Schjølberg, I. 2015a. A Risk Management Framework for Unmanned Underwater Vehicles Focusing on Human and Organizational Factors Proceedings of the ASME 2015 34th International Conference on Ocean, Offshore and Arctic Engineering OMAE2015, 31.05.-05.06.2015 St. John's, NL, Canada. ASME, pp. 1-10.

Thieme, C. A., Utne, I. B. & Schjølberg, I. 2015b. Risk Modeling of Autonomous Underwater Vehicle Operation Focusing on the Human Operator. *In:* Podofillini, L., Sudret, B., Stojadinovic, B., Zio, E. & Kröger, W., eds. 25th European Safety and Reliability Conference, ESREL 2015, 7.-10.09.2015 Zürich, Switzerland. Boca Raaton, London, New York, Leiden: CRC Press, Taylor & Francis Group, pp. 3653-3660

Tvete, H. A. 2015. Revolt; the Unmanned, Zero Emission, Short Sea Ship of Hte Future. *Green Ship Technology 2015.* Copenhagen, Danmark. pp. 1-18.

Utne, I. B. & Schjølberg, I. 2014. A Systematic Approach to Risk Assessment - Focusing on Autonomous Underwater Vehicles and Operations in Arctic Areas. Proceedings of the ASME 2014 33rd International Conference on Ocean, Offshore and Arctic Engineering, San Francisco, California, USA. pp. 1-10.

Utne, I. B., Sørensen, A. J. & Schjølberg, I. 2017. Risk Management of Autonomous Marine Systems and Operations. *Proceedings of the ASME 2017 36th International Conference on Ocean, Offshore and Arctic Engineering, OMAE 2017.* Trondheim, Norway. pp. 1-10.

Vagia, M., Transeth, A. A. & Fjerdingen, S. A. 2016. A Literature Review on the Levels of Automation During the Years. What Are the Different Taxonomies That Have Been Proposed? *Applied Ergonomics,* 53, Part A, pp. 190-202.

Valdez Banda, O. A. & Kannos, S. 2018. Hazard Analysis Process for Autonomous Vessels. Finnland: Aalto univeristy, NOVIA University of applied science. pp. 2-66.

Vinnem, J. E. 2010. Risk Indicators for Major Hazards on Offshore Installations. *Safety Science,* 48, pp. 770-787.

Wei, Y. Y. 2006. *A Study of Software Input Failure Propagation Mechanisms.* PhD Thesis, University of Maryland, College Park, MD.

Wei, Y. Y., Rodriguez, M. & Smidts, C. S. 2010. Probabilistic Risk Assessment Framework for Software Propagation Analysis of Failures. *Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability,* 224, pp. 113-135.

Whaley, A. M., Kelly, D. L., Boring, R. L. & Galyean, W. J. 2011. SPAR-H Step-by-Step Guidance. *Idaho National Laboratory Risk, Reliability, and NRC Programs Department Idaho Falls, Idaho.* Idaho Falls, Idaho, USA. pp. 1-18.

References

Woods, D. D. 2006. Essential Characteristics of Resilience. *In:* Hollnagel, E., Woods, D. D. & Leveseon, N. G. (eds.) *Resilience Engineering -Concepts and Precepts.* 1st ed. Surrey, UK; Burlington, USA: Ashgate. pp. 21-34

Wróbel, K., Krata, P., Montewka, J. & Hinz, T. 2016. Towards the Development of a Risk Model for Unmanned Vessels Design and Operations. *TransNav, the International Journal on Marine Navigation and Safety of Sea Transportation,* 10, pp. 267-274.

Wróbel, K., Montewka, J. & Kujala, P. 2017. Towards the Assessment of Potential Impact of Unmanned Vessels on Maritime Transportation Safety. *Reliability Engineering & System Safety*, pp. 155-169.

Wróbel, K., Montewka, J. & Kujala, P. 2018. System-Theoretic Approach to Safety of Remotely-Controlled Merchant Vessel. *Ocean Engineering,* 152, pp. 334-345.

Yamada, S. 2014. *Software Reliability Modeling - Fundamentals and Applications,* Tokyo, Heidelberg, New York, Dordrecht, London, Springer Japan.

Yan, R., Pang, S., Sun, H. & Pang, Y. 2010. Development and Missions of Unmanned Surface Vehicle. *Journal of Marine Science and Application,* 9, pp. 451-457.

Yuh, J., Marani, G. & Blidberg, D. R. 2011. Applications of Marine Robotic Vehicles. *Intelligent Service Robotics,* 4, pp. 221-231.

Zhu, D. 2005. *Integrating Software Behavior into Dynamic Probabilistic Risk Assessment.* PhD Thesis, University of Maryland, Collage Park, Md.

Zhu, D., Mosleh, A. & Smidts, C. 2007. A Framework to Integrate Software Behavior into Dynamic Probabilistic Risk Assessment. *Reliability Engineering & System Safety,* 92, pp. 1733-1755.

Øien, K. 2001a. A Framework for the Establishment of Organizational Risk Indicators. *Reliability Engineering & System Safety,* 74, pp. 147-167.

Øien, K. 2001b. Risk Indicators as a Tool for Risk Control. *Reliability Engineering & System Safety,* 74, pp. 129-145.

Øien, K. 2008. Development of Early Warning Indicators Based on Incident Investigation. 9th International Conference on Probabilistic Safety Assessment and Management PSAM 2008. pp. 1809-1816.

Øien, K. 2013. Remote Operation in Environmentally Sensitive Areas: Development of Early Warning Indicators. *Journal of Risk Research,* 16, pp. 323-336.

Øien, K., Massaiu, S. & Tinmannsvik, R. K. 2012. Guideline for Implementing the Rewi Method. Trondheim: SINTEF, IFE. pp. 1-36.

Øien, K., Massaiu, S., Tinmannsvik, R. K. & Størseth, F. 2010. Development of Early Warning Indicators Based on Resilience Engineering. 10th International Conference on Probabilistic Safety Assessment and Management, PSAM10. pp. 1762-1771.

Øien, K. & Paltrinieri, N. 2012. Resilience Based Indicators - Ability to 'Cope with the Unexpected' Resilience Based Early Warning Indicators - Complementary to Other Methods. SINTEF Technology and Society. pp. 6-52.

Øien, K., Utne, I. B. & Herrera, I. A. 2011. Building Safety Indicators: Part 1 - Theoretical Foundation. *Safety Science,* 49, pp. 148-161.

Part II – Articles

This page is intentionally left blank

Article 1

Thieme, C. A., Utne, I. B. & Haugen, S. 2018. *Assessing ship risk model applicability to marine autonomous surface ships, Ocean Engineering*, 165, pp. 140-145, DOI: 10.1016/j.oceaneng.2018.07.040.

This page is intentionally left blank

Ocean Engineering 165 (2018) 140–154

Contents lists available at ScienceDirect

Ocean Engineering

journal homepage: www.elsevier.com/locate/oceaneng

Assessing ship risk model applicability to Marine Autonomous Surface Ships

Christoph Alexander Thieme[a,b,*], Ingrid Bouwer Utne[a,b], Stein Haugen[b]

[a] *NTNU Centre for Autonomous Marine Operations and Systems (AMOS), Otto Nielsens veg 10, Trondheim 7491, Norway*
[b] *Department of Marine Technology, NTNU, Otto Nielsens veg 10, Trondheim 7491, Norway*

ARTICLE INFO

Keywords:
Marine Autonomous Surface Ship
Autonomous vessel
Collision risk
Allision risk
Grounding risk

ABSTRACT

Marine Autonomous Surface Ships (MASS) are tested in public waters. A requirement for MASS to be operated is that they should be at least as safe as conventional ships. Hence, this paper investigates how far the current ship risk models for ship-ship collision, ship-structure collision, and groundings are applicable for risk assessment of MASS. Nine criteria derived from a systems engineering approach are used to assess relevant ship risk models. These criteria aim at assessing relevant considerations for the operation of MASS, such as technical reliability, software performance, human-machine interfaces, operating, and several aspects of communication. From 64 assessed models, published since 2005, ten fulfilled six or more of these criteria. These models were investigated more closely. None of them are suitable to be directly used for risk assessment of MASS. However, they can be used as basis for developing relevant risk models for MASS, which especially need to consider the aspects of software and control algorithms and human-machine interaction.

1. Introduction

Marine Autonomous Surface Ships (MASS) are becoming increasingly interesting for the commercial maritime sector as an alternative to conventional ships. Several research projects have investigated MASS concepts (e.g., ReVolt; (DNV-GL, 2015); Maritime Unmanned Navigation through Intelligence in Networks (MUNIN, 2012); Advanced Autonomous Waterborne Applications (2016). Norway announced the first field test area for MASS, which is shared with public marine traffic (Norwegian Maritime Authority, 2016). The first autonomous cargo ship is supposed to be in operation by fall 2018 (Kongsberg Maritime, 2017).

A MASS may be low manned or unmanned (Rødseth and Nordahl, 2017), which creates challenges in operation. The MASS will influence risk in relation to several marine stakeholders, the environment, and the MASS itself. Collisions and groundings contribute most to the risk level for conventional ships (Pedersen, 2010). The MASS will be equipped with collision avoidance systems and sensory equipment for safe operation. Moreover, the MASS should at least be as safe as conventional ships (Advanced Autonomous Waterborne Applications, 2016; Nautilus Federation, 2018; Pedersen, 2010) to be acceptable for use in public ocean space.

Risk assessments serve to demonstrate a certain level of risk and are an important tool for making relevant design decisions (Rausand, 2011). Wróbel et al. (2017) assessed the effect of unmanned vessels and conclude that MASSs will reduce the collision frequency, while the severity of consequences might increase due to the reduced recovery capability. Hence, risk models, integrating technical, human, and organizational factors, are needed that reflect the operation of MASS. The Danish Maritime Authority (2018) has suggested adapting the international regulations such that MASS shall be developed following a goal- and risk-based regulatory approach.

Autonomous underwater vehicles (AUV) have been in the focus of risk research, such as risk management frameworks (Brito et al., 2012; Thieme et al., 2015a), and risk assessments (Brito and Griffiths, 2016; Brito et al., 2010; Griffiths and Brito, 2008; Thieme and Utne, 2017; Thieme et al., 2015b).

For MASS, less research has been conducted. Rødseth and Burmeister (2015b) and Rødseth and Tjora (2014) analyzed and presented the risk-based design methodology applied in the MUNIN project (MUNIN, 2012), which is based on the formal safety assessment (FSA) process of the International Maritime Organization (IMO, 2002).

The qualitative and quantitative analyses, including considerations of risk, of the MUNIN project were summarized by Kretschmann et al. (2015a,b). The detailed analysis of the MUNIN project was presented by Jensen (2015). Section 4 in the Advanced Autonomous Waterborne Applications (2016) white paper summarizes safety and security considerations and associated challenges for the development of MASS.

Wrobel et al. (2016) presented a Bayesian belief network (BBN) for assessing accidents for unmanned ships based on the mutual influence of different risk factors. Wróbel et al. (2018) developed a safety control structure model of MASS. It is analyzed with the System-Theoretic Process

[*] Corresponding author. Department of Marine Technology, NTNU, Otto Nielsens veg 10, Trondheim 7491, Norway.
E-mail address: Christoph.thieme@ntnu.no (C.A. Thieme).

https://doi.org/10.1016/j.oceaneng.2018.07.040
Received 6 February 2018; Received in revised form 6 June 2018; Accepted 12 July 2018
Available online 17 July 2018

Analysis (STPA), to identify possible scenarios where control structures may become inadequate. Both articles address the uncertainty in relation to MASS, their operation, and risk, which makes it difficult to develop a generic and comprehensive risk model for MASS.

The present article reviews selected grounding and collision risk models to identify practices and modelling approaches that may be applicable for risk modelling of MASS. It attempts to assess whether current collision and grounding risk models or parts of these can capture the unique aspects of MASS operation. A risk model for MASS operation needs to assess the level of risk, for example, the probability of ship collision.

The systems engineering process is used to identify criteria, which reflect aspects that should be represented in a risk model for MASS. The purpose is to identify potential gaps and focus areas that need to be especially addressed by new risk models developed for MASS.

Further this article focuses on operation of MASS (i.e., during transit in the oceans and seas), including vessel approaching ports or offshore installations. Vessels that are not in transit, which carry out specific tasks and operations (e.g., fishing vessels, offshore vessels moored, or in dynamic positioning mode, research vessels, military vessels, and other special purpose vessels) are excluded. Furthermore, security aspects are disregarded (i.e., the possibility of willful collision or grounding). Current international maritime legislation, such as the United Nations Convention of the Law of the Seas (UNCLOS, 1982), is not adapted to the advent of MASS. This aspect is disregarded in this article, assuming that conventional vessels and MASS are treated alike.

Models for detailed consequence analysis as part of risk assessment are not considered, only limited information on MASS concepts is available. To limit the scope of this article, only risk models that were developed since 2005 are considered in the article. The selected risk models assess the probability of ships colliding, stranding, and/or grounding.

A recent literature review by Lim et al. (2018) on maritime risk models summarizes the model types, modelling methods, and research contributions. Lim et al. (2018) identified future research directions in the maritime risk and security domain for conventional ships This current article is different from Lim et al.'s (2018), because this current article assesses possible modelling approaches from current risk models for conventional ships to MASS.

The next section presents the background and definitions. This is followed by the methodology. The criteria for the assessment of the risk models are identified in the section thereafter. The results section presents the findings and identifies gaps in the risk models that need to be addressed in future risk models for MASS. The models and approaches that are relevant for MASS are discussed in Section 6. This is followed by concluding remarks, and an outlook on further work.

2. Background

Risk models for ships are used to assess the risk arising from ship traffic, during ship operation, or for a marine area. Goerlandt and Montewka (2015) reviewed the use of risk definitions and quantification of risk of published maritime risk models. In many cases, these models do not state the risk definition or risk measure. A clear definition of the concept of risk and other related terms is necessary to clearly describe, communicate, and manage risk (Aven and Zio, 2014). In addition, the international maritime organization IMO (2002) defines risk for the framework of FSA as: "The combination of the frequency and the severity of the consequence." Moreover, SN-ISO Guide 73 (2009) defines risk as the "effect of uncertainty on objectives," whereas the effect can be positive or negative. Considering MASS, such a risk definition might be more suitable due to the expected uncertainties regarding the technical solutions, operation, and environment.

2.1. Autonomy and Marine Autonomous Surface Ships

Autonomous systems may have different levels of autonomy (LoA). Autonomy is a system's ability to make independent decisions from a supervising agent and execute these decisions (Vagia et al., 2016). For conventional marine vessels, the supervising operators are the crew. For MASS, only one or a few operators will take a supervising role and intervene when necessary. This is described in more detail in Section 2.2.

The LoA describes the degree of this ability to make independent decisions (Vagia et al., 2016). Typically applied LoA scales are presented by Sheridan and Verplank (1978) or Endsley and Kaber (1999). Comprehensive reviews are provided by Insaurralde (2012) or Vagia et al. (2016). Rødseth and Nordahl (2017) and Utne et al. (2017) defined each specific scale for MASS with four levels. These scales define the decision authority and the tasks that the human operators and the autonomous system carry out, implicitly affecting risk. In this case, the term *tasks* refers to information acquisition, information analysis, decision selection, and action implementation (Parasuraman et al., 2000). In the lowest LoA (i.e., manual control (Endsley and Kaber, 1999; Vagia et al., 2016) the human operator does everything, and the autonomous system does not assist.

In intermediate LoAs, the autonomous system and the operators cooperate (Endsley and Kaber, 1999; Rødseth and Nordahl, 2017; Utne et al., 2017). In the highest LoA (full autonomy), the human operator has no possibility to intervene with the system (Endsley and Kaber, 1999; Rødseth and Nordahl, 2017; Sheridan and Verplank, 1978; Utne et al., 2017). This is not likely for MASS, at least in the near future.

Autonomy and automation are used often interchangeably, although different aspects are included in the concepts (Vagia et al., 2016). The term *autonomy* will be solely used in this article. An autonomous system capable of changing the LoA according to the circumstances is designed with adaptive autonomy (Sheridan, 2011).

2.2. Operation of conventional versus autonomous ships

No formal definition of a conventional ship exists. The UNCLOS (1982) does not define a ship or vessel (Danish Maritime Authority, 2018). Therefore, information on common practices is used. A ship or vessel has a crew for the engine department, the bridge, the deck department, and stewards. The crew level of a cargo ship ranges between ten and 21 people (Curley, 2012). The master of a vessel has the responsibility for the vessel, its safety, personnel, cargo, and passengers. The master has the aboard decision authority. The master acts as a communication point between the shipping company, crew, and other actors (Norwegian Shipowners' Association, 2003). The bridge crew is responsible for navigation and control over the ship. Moreover, UNCLOS (1982) requires a lookout at all times, according to the conditions, and that communication via radio is maintained. The bridge must be staffed according to weather and visibility conditions. A voyage plan must be determined and approved by the master before the vessel sets sail (Norwegian Shipowners' Association, 2003; UNCLOS, 1982).

The chief officer is responsible for the navigation and is second in command. Mates and able sea folk act as lookouts. The deck crew handles the cargo and loads and offloads the vessel (Norwegian Shipowners' Association, 2003). The stewards are responsible for crew well-being. The engine department is responsible for supervision and preventive and corrective maintenance of the machinery (Curley, 2012). The chief engineer is responsible for the engine department (Norwegian Shipowners' Association, 2003).

Rødseth and Burmeister (2015a) and Advanced Autonomous Waterborne Applications (2016) showed that there will be several technical solutions for MASS. The MASS need to be designed for their purpose with different performances, advantages, and disadvantages. Three main concepts of operation of autonomous ships can be differentiated: (i) MASS with low manning (Bertram, 2016), (ii) "master slave" supervision (Bertram, 2016), and (iii) shore control center (SCC) supervised MASSs (MUNIN, 2012; Rødseth and Nordahl, 2017; Rødseth et al., 2014). The main difference in these concepts is the location of the operators or supervisors since none of these concepts are fully autonomous. Current concepts rely on an operator with decision authority supervising the MASS. The operational concepts can only be described superficially, since they depend on the size and purpose of the vessel (Advanced Autonomous Waterborne Applications, 2016).

The three concepts mentioned above all have a control system of the

MASS that collects information on the environment, analyses it, makes decisions based on these analyses, and acts accordingly. The MASS needs to be able to sense the environment through machine vision and sensor fusion, for example (Advanced Autonomous Waterborne Applications, 2016; Bertram, 2016). The operators have a supervisory role during voyages and can take control of the MASS when necessary (e.g., if several obstacles are detected, in dense traffic, or during port approach). The operators also handle necessary radio communications with other vessels or vessel traffic service (VTS).

The MASS with low manning (i) are an intermediate solution during the transition period to autonomous vessels that are unmanned (Bertram, 2016; Kongsberg Maritime, 2017). The crew on board a vessel is then reduced in comparison to conventional shipping. The crew can perform necessary maintenance and take control of the MASS if necessary. The MASS will be mostly in autonomous mode and does not require operator input.

In the “master slave” supervision system (ii), one manned vessel supervises several unmanned vessels. All vessels travel together, and the crew of the manned vessel can take control of the unmanned vessels if necessary. Near ports, pilots and tug boats might assist the vessels (Bertram, 2016). Maintenance of components is, in this concept, rather limited during the voyage, and advanced monitoring systems are needed.

A SCC supervised MASS (iii) configuration (MUNIN, 2012; Rødseth and Nordahl, 2017; Rødseth et al., 2014) is not manned during voyage and is remotely supervised from a land-based SCC. The SCC communicates with the MASS through satellites or through radio-based systems, when the MASS is near the shore. The MUNIN project envisions that, for entering ports, a crew boards the vessel and takes manual control over the vessel (Rødseth et al., 2014). The ReVolt concept envisions low aid needed, through adapted port design and new docking technology (Tvete, 2015). Since MASS that are supervised by a SCC are mainly unmanned, the opportunities for maintenance are limited. Preventive and corrective maintenance can only be executed during port time or dry docking (Rødseth and Burmeister, 2015a). This demands a highly reliable system and proactive condition monitoring that identifies incipient failures. Bertram (2016) argued that conventional diesel engines might not be suited for unmanned shipping since they need frequent maintenance. New concepts, such as hydrogen or battery driven propulsion, are needed (Bertram, 2016; Tvete, 2015).

3. Method

3.1. Selection of risk models

This article considers only models developed since 2005. The MASS concept has received increased attention in recent years due to technical availability and expected financial feasibility. Only models that assess the risk associated with collisions, allisions, or grounding are considered. Allisions are ship-structure collisions (Hassel, 2017; Hassel et al., 2017). The Scopus[1] database was searched for the keywords: “Ship OR vessel AND collision model,” “Ship OR vessel AND Allision,” and “Ship OR vessel AND grounding OR stranding model.” The search was conducted on November 3, 2017. Additionally, publications referenced in the literature were included, if possible. Additional references were found in work by Goerlandt and Montewka (2015). One master thesis and one doctoral thesis were included that were not listed in Scopus or by Goerlandt and Montewka (2015): Jensen (2015), and Hassel (2017). Three publications that address MASS, are included. These are Jensen (2015), Wrobel et al. (2016), and Wróbel et al. (2018).

Models that do not give enough information on how the frequency or probability were assessed have been excluded. In accordance with the scope, models covering inland waterways, rivers, or arctic areas have been excluded, such as those by Almaz (2012) or Zhang et al. (2013). Similarly, Valdez Banda et al. (2015) presented a model for risk assessment in ice operation, which resembles a special operation.

Johansson and Molitor (2011) presented a risk assessment for the Baltic Sea, reusing existing models and software. Goerlandt et al. (2012) presented a holistic risk assessment based on previously defined risk models by Hänninen and Kujala (2010) and Goerlandt and Kujala (2011). These three models are assessed as one model in the analysis since they build upon each other.

3.2. Development of assessment criteria

To identify suitable and relevant criteria for assessing the existing risk models, a systems engineering approach is employed. First, the problem and the desired systems are described (i.e., the MASS operation). This is the first phase of a systems engineering process (Blanchard, 2008). In the second step, system requirements are described and functional needs with respect to safety are identified. Typical questions answered in the requirement identification are as follows (Blanchard, 2008):

1. What is required from the system, stated in functional terms?
2. What specific functions must the system accomplish?
3. What are the primary functions to be accomplished?
4. What are secondary functions to be accomplished?
5. What must be accomplished to completely alleviate the stated deficiency?
6. Why must these functions be accomplished?
7. When must these functions be accomplished?
8. Where is this to be accomplished and for how long?
9. How many times must these functions be accomplished?

Not all of these questions can be answered in this article. However, they are used as guidelines for the identification of the needs and requirements for MASS. These give input to the identification of suitable assessment criteria.

3.3. Assessment procedure

The identified relevant ship risk models are categorized according to their approach to risk assessment. The approaches are generally discussed for their applicability and possible further use for MASS. The identified models are assessed against the criteria from Table 2 in Section 4.2.

The models that fulfil most of the criteria are further analyzed in Section 6. Models that fulfill several criteria, are assumed to reflect a high level of detailed modeling of the interaction between the risk relevant modelling aspects summarized in the criteria. The suitability of the models and possible learnings from these are highlighted. This does not imply that the models may be used as they are but they may be used as basis for developing MASS specific risk models.

4. Evaluation criteria

4.1. Functional requirements with respect to risk

The main function of MASS is to transport goods or people from one port to another. This is the same main function as for conventional ships. The transport needs to be safe, cost efficient, and reliable. The main difference between MASS and conventional ships is the reduced crew, which may have implications for the design of the vessels. Safety related functions currently executed by the crew must be carried out by the MASS and its subsystems. The functions in relation to safety are situational awareness of the environment and the surroundings of the vessel, which is the task of the lookout and the purpose of the navigational systems (e.g., RADAR) on a conventional vessel. A more detailed functional analysis and description for autonomous ships can be found in the work by Rødseth and Nordahl (2017).

Table 1 summarizes the requirements for MASS that follow from the description in the previous section. The MASS should identify obstacles and potential hazards and react appropriately in a timely manner (R1). Sensors, computers, and actuators need to execute these functions in a reliable

[1] www.scopus.com, accessed on Nov. 3, 2017.

Table 1
Requirements for MASS based on the operational differences for conventional vessels, identified through an adapted systems engineering process.

Requirement	Description
R1	Reliable and timely identification of obstacles and hazards
R2	Reliable MASS during voyage (sensors, machinery, and control system)
R3	Robust and verified software and algorithms
R4	Reliable communication lines between MASS and the control basis for remote supervision and operation
R5	Reliable and adequate communication among operators and crew
R6	Reliable and adequate communication between MASS operators and other marine stakeholders
R7	Accessible and affordable human-machine interfaces
R8	Adequate provisions for adaptive autonomy

manner, and they need to be available during the voyage. The opportunities for maintenance and repairs are limited. The MASS need to be reliable with respect to sensor systems, machinery, and the control system to achieve their mission goals (R2). The software side and algorithms need to be robust, and verification of their safe performance is desirable (R3). Due to the natural differences between software and hardware, different methods for risk assessment of these are needed (Leveson, 2011).

Current concepts for MASS (i to iii) still rely on human operators to some degree, partly on board the MASS. They supervise the MASS, adapt the mission plan, or take over control if necessary. Concepts ii and iii require that reliable communication lines with sufficient transmission capacity exist between the MASS and the operators, such that safe operation is possible (R4). There is need for suitable provisions for a crew since it might be necessary to board the ship for berthing (MUNIN, 2012; Rødseth and Nordahl, 2017).

Two more types of communication need to be considered: reliable and adequate communication among the crew/operators in the SSC or on board a low manned vessel in situations that require the human operators to intervene (R5) and communication between MASS operators and other ships or VTS (R6). Both types of communication need to be unambiguous and goal oriented to ensure safe operation. The MASS should be easily accessible for the operators through the provided user interfaces (R7). The operators need to be able to assess the present situation quickly to develop a good situation awareness and be able to reason about necessary actions. Hence, human-machine interfaces (HMI) need to be optimized for usability and accessibility. In cases in which the operators take control of the MASS, the LoA will change, which is called adaptive autonomy (R8). The system and operators must be able to adapt quickly to the new operational mode with a different LoA.

4.2. Evaluation criteria

Based on the previously identified requirements (cf. Table 1), the criteria for evaluating the risk models are derived. The criteria reflect the needs of a MASS (i.e., what aspects a ship risk model should cover to be suitable for MASS). Table 2 summarizes the identified criteria for risk model evaluation. It is not possible to rank the importance of these criteria, since each criterion covers important aspects of risk modelling for MASS that need to be included in a risk model.

Criterion 1 summarizes the main difference between MASS and conventional ships. MASS operation will to a large degree depend on software functionality. Autonomous functions, control algorithms, and other software aspects that are failing influence risk.

MASS operation may require a substantial amount of interaction between the MASS and its operators during parts of the voyage. Therefore, it is necessary to consider the HMI and the operators' interaction with the HMI (C2). Communication is also an important aspect in the cooperation and interaction between actors. The operators of one vessel (mainly concepts i and ii) need to communicate to detect and resolve hazardous situations (C3).

Criterion 4 investigates remote communication with the shore base. Conventional vessels should receive substantial support from the shore organization (Norwegian Shipowners' Association, 2003), which requires robust communication lines with the SCC. The MASS may be monitored from a SCC (concept iii), which requires that remote connections are considered. The MASS operating with concepts i and ii might have less contact with the SCC.

MASS may be unmanned and it may not be possible to perform maintenance immediately when necessary. This is especially true for long voyages. Hence, the system reliability and maintenance (C5), and backup solutions in case of failure of a sub-system (through functional redundancy, C6) are important. A risk model should consider functional redundancies that were introduced in the system to reflect the risk level accurately. The MASS will employ several sensor systems to create a holistic operational picture via, for example, sensor fusion.

Criterion 7 aims at the assessment of the models with respect to different operational modes and LoA, such as piloted, auto-piloted, manual control, or autonomous voyage. Consideration of the operational mode is necessary since the operators' interaction with the vessel and the performance of the vessel itself will change. The vessel navigation will vary in these modes.

Criterion 8 assesses whether the risk models consider communication between the vessel crew and other marine participants, such as other ships or manned structures.

Criterion 9 assesses whether the risk models include considerations of personnel (e.g., different manning levels, different roles on board the ship, and operating the vessel). This is closely connected to the operational mode and LoA (C1) and communication aspect between operators (C4). The crew level (C9) dependends on the operational concept and may not be relevant for complete unmanned systems. However, it is important for low manned or partially unmanned systems.

5. Results

Table 3 summarizes the 64 reviewed models with the following information: accident type, object of analysis, model aim, modeling methods, model parameters, and data sources. With respect to the type of accident, 14 models cover collision and grounding, seven models

Table 2
Identified evaluation criteria for ship risk model evaluation for adaptability to MASS.

No.	Criterion	Addressed Requirements from Table 1
C1	Inclusion of software and control algorithm performance	R3, R7
C2	Inclusion of human-machine interfaces and ergonomic considerations	R7
C3	Inclusion of communication between vessels and shore base	R4
C4	Inclusion of communication between operators	R5
C5	Inclusion of aspects of maintenance and reliability of system performance	R1, R2
C6	Inclusion of functional redundancy	R1, R2
C7	Consideration of different operational modes and change of LoA	R8
C8	Inclusion of communication between operators and other marine participants	R6
C9	Consideration of different crew levels	R2, R8

Table 3

Characteristics of the reviewed risk models. Abbreviations: accident types: CG – collision and grounding, G – grounding/stranding, SSC – ship-ship collision, A – allision; object of analysis: MTS – maritime transportation system, S – ship, ST – ship type, W – waterway; modelling techniques: AHP – analytical hierarchy process, BBN – Bayesian belief network, BT – Bayesian theorem calculations, ETA – event tree analysis, F – fuzzy inference, FMEA – failure mode effect analysis, FTA – fault tree analysis, GM – geometrical formulation, R – regression model, Sim – simulation, STPA - system theoretic process assessment; data source: AD – accident data, EJ – expert judgement, HD – historical data, PD – published data, RT – real-time data.

Model	Reference	Accident Type	Object of Analysis	Model Aim	Modelling Techniques	Parameters in the Model	Data Source
M1	Merrick and van Dorp (2006)	SSC	W, ST	Framework for risk and uncertainty assessment in maritime systems.	BT, Sim	Propulsion failure, steering failure, navigational aid failure, human error, error by a nearby vessel, visibility, weather, and fairway characteristics	AD, EJ, HD
M2	Hu et al. (2007)	CG	W	Assess risk of piloted vessels in a harbor.	F	Observed frequencies of accidents, traffic flow, vessel traffic characteristics, and fairway characteristics	AD, HD
M3	COWI (2008)	CG	W	Assess the effects of waterway separation measures on the risk level.	GM	Traffic flow, vessel traffic characteristics, local experience, pilotage, safety standards, and fairway characteristic	EJ, HD, PD
M4	Ellis et al. (2008)	A	W	Assess effects of windfarms on the risk level in a waterway.	GM	Traffic flow, vessel traffic characteristics, fairway characteristics, technical failure, external assistance, self-repair of technical failure, fail to anchor, vessel motion model, failure of navigational equipment, human error, weather, visibility, failure to warn vessel on collision course, and crew reaction time	HD, PD
M5	IWRAP (described by Friis-Hansen (2008))	CG	W	Framework to assess the risk in a waterway and decide on risk reduction measures.	BBN, GM	Traffic flow, traffic vessel characteristics, fairway characteristics, weather, RADAR performance, daytime, stress, alarms, officer of the watch (OOW) training and vigilance, propulsion failure, repair time, and bridge design	HD, PD
M6	Przywarty (2008)	G	W	Model to assess the grounding risk in a waterway and assess risk reduction measures.	BT, FTA, GM, Sim	Human error, sensor errors, position estimation/measurement error, disuse of information, failure to use assistance, insufficient assistants provided, no/delayed assistants, maintenance errors, environmental constraints, material failure, inability to repair, unsafe winds and currents, vessel characteristics, and topography	HD, PD
M7	Trucco et al. (2008)	SSC, A	MTS	Framework to assess the risk in a waterway and decide on risk reduction measures.	BBN, FTA	Crew and personal characteristics, compliance with rules, climate, automation and mechanical failures, maneuvering errors, traffic density, visibility, weather, sea state, and influences from operating organization	EJ, HD
M8	Wang and Fan (2008)	SSC	W	Assess the risk in a waterway and identification of risk reduction measures.	Sim	Traffic flow, vessel traffic characteristics, fairway characteristics, safety regulations, visibility, and wind	AD, HD
M9	Chin and Debnath (2009)	SSC	W	Collision warning system for pilots.	R	Vessel size, day time, time to accident, and distance to accident	EJ, RT
M10	Debnath (2009)	SSC	W	Collision warning system for pilots and VTS agents.	GM	Number of possible interactions, day/night time, ship density, vessel traffic characteristics, and waterway characteristics	EJ, HD, PD
M11	Klemola et al. (2009)	SSC	W	Framework to assess the risk in a waterway and decide on risk reduction measures.	BBN, GM	Traffic flow and causation probability including human factors	EJ, HD, PD, RT
M12	Martins and Maturana (2009)	SSC	W, ST	Incorporate human performance in risk assessment and assessment of risk mitigation for tankers.	BBN	Communication on bridge, communication with other vessel, human error of master and nautical officer, detection failure, wrong information available, failure in navigational planning, weather, sea state, visibility, concentration, personal factors, workload, RADAR detection, and alarm detection	EJ, PD
M13	Ozbas et al. (2009)	SSC	W	Assess the risk in a waterway and decide on risk reduction measures.	Sim	Traffic flow, vessel traffic characteristics, fairway characteristics, vessel reliability, technical failure, communication/navigational aid failure, request for pilot or tugboat, visibility, current, hourly traffic variations, and fairway complexity	EJ, HD
M14	Uluscu et al. (2009)	CG	W	Assess the risk in a waterway and decide on risk reduction measures.	Sim	Traffic flow, vessel traffic characteristics, human error, steering failure, propulsion failure, communication/navigational equipment failure, mechanical/electrical failure, tugboat/pilot assistance, visibility, currents, and day time	EJ, HD
M15	Vanem et al. (2009)	CG, A	S	Generic standardized risk model for different ships following FSA procedure	Suggest BBN, FTA	Collision/grounding/contact frequency model, flooding frequency model, survivability, model, time to sink model, evacuation model, environmental damage model	–

(*continued on next page*)

Table 3 (*continued*)

Model	Reference	Accident Type	Object of Analysis	Model Aim	Modelling Techniques	Parameters in the Model	Data Source
M16	COLWT (described by Povel et al. (2010)	A	W	Assess effects of windfarms on the risk level in a waterway with a developed framework and risk acceptance criteria.	BBN, GM	Human error to avoid collision, technical failure, visibility, weather, sea state, RADAR status, and AIS functionality	HD, PD
M17	Debnath and Chin (2010)	SSC	W	Framework to assess the risk in a waterway and identify vessel types with the highest risk level for VTS and harbor authorities.	GM	Proximity indicators, vessel characteristics, day time, and scenario dependent collision probability	EJ, HD
M18	Kaneko (2010)	G	W	Framework to assess the grounding risk in a waterway and assess risk reduction measures.	F, GM	Two approaches with similar characteristics: vessel traffic characteristics, traffic flow, fairway characteristics, position fixing time, and omission probability,	HD, PD
M19	Montewka et al. (2010)	SSC	W	Framework to identify potential collision candidates.	GM, Sim	Detailed vessel traffic characteristics (length, draft, resistance, thrust, maneuverability), and vessel motion models, season	HD, PD
M20	Montewka et al. (2011)	CG	W	Assess collision risk in a waterway.	BBN, GM, Sim	Traffic flow, seasonal/daily/hourly variations in traffic flow, vessel traffic characteristics, vessel motion model, human error, technical failure, technical equipment available (grounding), and channel characteristics	HD, PD
M21	Ren et al. (2011)	SSC	W	Collision warning system for ship navigators, pilots and VTS agents.	F, GM, Sim	Distance to closest point of approach, time to closest point of approach, traffic flow, and encounter angle	HD, RT
M22	van Dorp and Merrick (2011)	CG	W	Assess the risk in a waterway and decide on risk reduction measures.	BT, GM, Sim	Traffic flow, weather, sea state, visibility, technical failure, human error, navigational aid failure, pilotage/towing, assistance from of VTS, increased surveillance, bridge alarms, company policies, training, and traffic rules,	EJ, HD
M23	Yang et al. (2011)	G	S	Collision warning system for pilots and VTS agents.	F, GM, Sim	Vessel characteristics and waterway characteristics	RT
M24	BRISK (described by COWI (2012))	CG	W	Assess traffic development, the risk in a waterway and identification of risk reduction measures.	GM, Sim	Traffic flow, vessel traffic characteristics, seasonal variations, human and technical failure, effect of implemented risk-reducing measures, and training	HD
M25	Goerlandt et al. (2012), Goerlandt and Kujala (2011), (Hänninen and Kujala, 2010)	SSC	W, ST	Assess traffic development, the risk in a waterway and identification of risk reduction measures.	BBN, Sim	Traffic flow, vessel traffic characteristics, weather, visibility, monthly/daily/hourly variations, technical reliability, management factors, human factors, support from VTS, and pilotage	AD, EJ, HD, PD
M26	Jeong et al. (2012)	SSC	W	Collision warning system for pilots and VTS agents.	GM	Closest point of approach, time to closest point of approach, traffic flow, vessel traffic characteristics, visibility, and weather	AD, HD
M27	Kaneko (2012)	G	W	Assess the grounding frequency.	GM	Fairway characteristics, traffic flow (assessed with two different methods) omission error, time fixing interval, length and time of a course trajectory, and vessel traffic characteristics	AD, HD
M28	Montewka et al. (2012a)	SSC	ST	Risk assessment of liquefied natural gas tankers with tugboats.	BBN	Technical failure, human error, weather, distance between vessels, and number of tugs	EJ, PD
M29	Montewka et al. (2012b)	SSC	W	Assessment of collision candidates and collision probability.	GM, Sim	Ship type, maneuverability, intersection angles, and maneuvering patterns	HD
M30	ShipRisk (described by Rasmussen et al. (2012)	CG	W	Assess effects of constructions on the risk level in a waterway and assessment of mitigation measures.	GM	Traffic flow, waterway characteristics, vessel traffic characteristics, human failure (navigation, conducting evasive maneuver), technical failure (loss of propulsion, loss of steering), and repair probability	AD, EJ, HD, PD
M31	Suman et al. (2012)	SSC	W	Assess effects of fairway rules' effect on the risk level in a waterway.	GM	Traffic flow, traffic vessel characteristics, closest distance to approach, time to closest distance of approach, visibility, and hourly variations	HD, RT
M32	Weng et al. (2012)	SSC	W	Risk assessment of a waterway and identification of mitigation measures.	GM	Traffic flow, traffic vessel characteristics, and time of day	HD, PD
M33	Blokus-Roszkowska and Smolarek (2013)	SSC	W	Assess the probability of collision and suggest traffic separation schemes.	GM, Sim	Traffic vessel characteristics, traffic flow, probability of giving way to another vessel, and traffic rules	HD, PD
M34	Silveira et al. (2013)	SSC	W	Risk assessment and identification of mitigation measures in a waterway.	GM	Traffic flow and vessel traffic characteristics	HD, PD
M35	Xiao et al. (2013)	SSC	W	Assess effects of constructions on the risk level in a waterway and assessment of mitigation measures.	Sim	Vessel traffic characteristics, traffic flow, collision avoidance maneuvers, weather, sea state, and COLREG	HD

(*continued on next page*)

Table 3 (*continued*)

Model	Reference	Accident Type	Object of Analysis	Model Aim	Modelling Techniques	Parameters in the Model	Data Source
M36	Akhtar and Utne (2014)	G	ST	Framework to assess the influence of factors influencing fatigue on the risk level on a tanker. Predict crew performance and effects of mitigation measures.	BBN	Vessel types and characteristics, organizational influences, manning, safety culture and climate, work scheme, procedures, qualifications and certifications, communication, fatigue, season, type of fairway, weather, sea state, visibility, human error, and failure	AD, HD, PD
M37	Burmeister et al. (2014)	SSC – vessels anchoring	W	Risk assessment and identification of mitigation measures in a waterway with anchoring vessels.	GM	Vessel traffic characteristics, traffic flow, vessels at anchorage, weather, and tidal currents	HD, PD
M38	Collide (described by Hassel et al. (2014))	A	W	Identification of improvements to the Collide risk assessment model, which assesses the allision risk level of offshore installations and suggests mitigation measures.	FTA	Traffic flow, unawareness of installation, no evasive maneuver planned, evasive maneuvers, human failure (navigation, watch keeping) equipment and technical failure, and failure of the installation initiating a recovery	EJ, HD
M39	Khan et al. (2014)	SSC, A	W	Framework for oil tanker risk assessment in arctic waters.	BBN	Human error, speed, equipment error, technical failure, visibility, weather, sea state, and ice conditions	EJ
M40	Montewka et al. (2014)	CG	W	Framework for grounding risk assessment, highlighting the validation and verification process.	BBN	Noise, vessel motion, vibration, stress, maintenance, technical failure, absence of personnel from bridge, and other vessel evasive actions	EJ, PD
M41	Mulyadi et al. (2014)	SSC	W	Assess the risk arising from ships colliding and sinking over a gas pipeline.	BBN, GM	Traffic flow, vessel traffic characteristics, probability to be over a pipeline, human performance, weather, visibility, navigational aid detection, pilotage, communication with other vessels, and steering failure	HD, PD
M42	Tvedt (2014)	A	W	Framework for allision with offshore structures.	ETA, FTA, BBN	Presence of officer on bridge, human failure (lookout, steering, setup navigational equipment, technical failure, navigational systems, steering), crew characteristics, workload, communication, task management, safety culture, management policies, reliance on technical equipment, bridge layout, roles and responsibilities, visibility, HMI, reliability and condition of equipment, manning, procedures, and system feedback	AD, HD, PD
M43	Zaman et al. (2014)	SSC	W	Risk assessment and identification of mitigation measures in a waterway.	F, FMEA	Traffic flow, vessel traffic characteristics, human error, weather, daily variations, and failure of machinery and electricity	EJ, HD, PD
M44	Goerlandt et al. (2015)	SSC	W	Collision warning system for ship navigators.	AHP, F, GM	Distance and time to closest point of approach, distance between vessels, traffic flow, bearings, reaction time, vessel type, visibility, time of day, sea state, and maneuvers of the vessels	EJ, HD, RT
M45	Jensen (2015)	SSC	W, ST	Risk assessment for an unmanned bulk carrier on a route.	ETA, FTA, GM	Traffic flow, human error, visibility, weather navigation system, software failure, communication, reliability of machinery, engine and propulsion system, COLREG rules, and manned/unmanned operation	EJ, HD, PD
M46	Khaled and Kawamura (2015)	SSC	W	Risk assessment and identification of mitigation measures in a waterway.	BBN, GM	Visibility, weather, daylight, familiarity with the location, navigational aids, communication, fairway characteristics, fairway markers, crew characteristics, reliability of steering equipment, engine technical condition, pilotage, manning, lookout, sea state, country of origin, COLREG rules, traffic flow, and traffic vessel characteristics	EJ, HD, PD
M47	Przywarty et al. (2015)	SSC	W, ST	Risk assessment and identification of mitigation measures in a harbor entrance.	GM, Sim	Vessel traffic characteristics and traffic flow	AD, HD
M48	Zhang et al. (2015)	SSC	W	Assessment of risk level in a waterway using risk indicators. Identification of encounter situations and risk mitigation measures.	GM	Distance, encounter angle, and relative speed between ships	EJ, HD
M49	Copping et al. (2016)	CG	W	Assess effects of windfarms on the risk level in a waterway.	GM, Sim	Traffic flow, vessel traffic characteristics, vessel behavior, seasonal variations, weather, currents, and vessel traffic rules	EJ, HD
M50	Ma et al. (2016)	A	W	Collision warning system for RADAR operators.	BBN, artificial potential fields	Traffic flow, velocity, size, and authenticity (of warning)	RT

(*continued on next page*)

Table 3 (*continued*)

Model	Reference	Accident Type	Object of Analysis	Model Aim	Modelling Techniques	Parameters in the Model	Data Source
M51	Mazaheri et al. (2016)	G	MTS	Generic grounding risk assessment for ships and decision-making. Use of strength of knowledge.	BBN	Bridge resource management, Safety culture, manning, communication, visibility, weather, preparation, bridge design, situational awareness, training and competence, maintenance, technical redundancy, VTS, pilotage, season, traffic distribution, adequate alarms, waterway complexity, technical failure, and navigation method	AD, HD, PD
M52	Norwegian National Ship Risk Model (Nilsen (2016) and Haugen et al. (2016))	CG	W, S	Model for risk assessment and decision support for maritime regulation and management in Norwegian waters.	BBN	Regulations and policies, fairway characteristics, external navigational aids, market and economic conditions, work organization, human resource management, manning level, social measures, education and training, safety management system, organizational model, maintenance, resource management, crew characteristics, ship characteristics, communication, task load, bridge design, navigational system design, technical condition of navigational aids, propulsion system, steering system, and communication system (external)	EJ, HD, PD
M53	Nivolianitou et al. (2016)	CG	W	Risk assessment of waterways, highlighting the ships with the highest contribution.	BBN	Ship type, flag state, ship age, and ship size	HD
M54	Rekha et al. (2016)	G	W	Grounding candidate identification for a waterway.	GM, Sim	Time, date, vessel types, cause of incidents, and weather	AD, HD
M55	Senol and Sahin (2016)	CG	S	Online risk assessment for ships, supporting navigators of the vessel.	FTA	Machinery failure, steering failure, failure in voyage planning, external failure (Tug, communication between vessels, etc.), perception failure, human error, weather, and lack of communication	EJ. RT
M56	Sotiralis et al. (2016)	SSC	S	Collision risk assessment and mitigation measure assessment for generic ships considering the human operators.	BBN	Bridge layout, human error, performance of OOW, navigational equipment error, non-bridge equipment failure, communication bridge crew, training, personal factors, and organizational factors, external communication	EJ
M57	Wrobel et al. (2016)	CG	S	Develop a risk model for unmanned vessel operation and design	BBN	Propulsion, steering, electrical power, other systems, communication, maintenance regime, sensors' performance, control algorithms, external information, alerting	–
M58	Afenyo et al. (2017)	A	W, S	Generic allision risk model for arctic waters for decision-making.	BBN	Iceberg presence, RADAR error, visibility, weather, human error, steering failure, propulsion failure, communication equipment failure, miscommunication, and navigational equipment failure	HD, PD,
M59	Chai et al. (2017)	SSC	W, S	Simulation based risk assessment and identification of mitigation measures in a waterway.	GM, Sim	Causation probabilities for different weather/visibility conditions, ship type, traffic flow, and minimum distance to collision	EJ, PD
M60	Hassel (2017)	A	W	Risk assessment for waterways with an offshore installation.	BBN	Passing distance of vessel, communication with ship, weather, communication equipment, visibility, bridge ergonomics, navigator skills, manning, alert systems, voyage planning, loss of power, loss of steering, performance of RADAR organizational factors, vessel characteristics, manning, traffic surveillance, and navigational equipment performance	EJ, HD
M61	Huang et al. (2017)	SSC	W, S	Online risk assessment for ships, supporting navigators of the vessel.	GM	Vessel velocity and acceleration data, vessel behavior, and vessel characteristics	EJ
M62	Presencia and Shafiee (2017)	A	W, ST	Risk assessment and identification of mitigation measures of vessels navigating to offshore windfarms.	GM	Traffic flow, vessel characteristics, and collision ratio	EJ
M63	Khan et al. (2018)	A	W, S	Generic allision risk model for arctic waters for decision-making.	BBN	Ice parameters, knowledge of crew, communication of crew, fatigue, human error, navigational equipment failure, weather, visibility, radio communication, voyage planning, safety measures, and ship class	EJ
M64	Wróbel et al. (2018)	CG, A	S	Analyze how different systems elements may lead to hazards for unmanned ships	STPA	Organizational environment, shore facilities (operator, company managers, passage plan, alarms), communication, vessel (internal sensors, on-board control system, auxiliary systems, engine, rudder, environmental sensors), navigation, environment (other ships, global navigation system),	–

focus on grounding or stranding (seven models), 28 models cover ship-ship collision, and nine models cover allision. Three models include both ship-ship collision and allision.

The object of analysis refers to the target of the risk assessment. These are general maritime transportation systems (referring to any of the following systems), certain ship types, or specific waterways. Most reviewed models aim at risk assessment for a certain region or waterway (43). Six models aim at a specific ship type in a waterway (e.g., ferries in a harbor area (M1) or specific oil tanker traffic areas (M12, M25, and M47). Models for specific vessels are presented for generic maritime transportation systems (M7, M15 and M51), for general cargo ships (M23 and M36), and autonomous vessels (M45, M57, M64). The models addressing MASS are described and discussed in more detail in Section 5.

Most models aim to assess the risk level in a waterway and assess the effect of risk-reducing measures, such as adapted traffic schemes and patterns. Some of these consider the change of the risk level through obstructions or structures, such as anchoring vessels (M36), bridges and structures (M4, M16, M46, and M62), offshore oil and gas platforms (M38, M44, and M60), or wind parks (M30 and M35). Only one model aims at the risk assessment of an MASS on a certain route, assessing the potential encounter frequency and probability of collision (M45).

The most commonly used modeling techniques and assessment approaches used in the risk models are geometric models (35 models), BBNs (24 models), and simulations (18 models). Less-used methods include the analytical hierarchy process (AHP, one model), Bayesian theorem calculations (three models), fuzzy inference (six models), event tree analysis (ETA, three models), failure mode and effect analysis (FMEA, one model), Fault tree analysis (FTA, 7 models), regression modeling (one model), and STPA. For detailed description of these methods, the reader is referred to the respective literature.

Data sources refer to the input for modeling and quantification of the models. Most models use historical data (48 models), expert judgment (31 models), published data (30 models), or a combination of these. Few models are not quantified, due to their generic nature or the modelling approach (M15, M57, and M64).

Historical data includes information obtained through automatic identification system (AIS) data, VTS, or other records of shipping information. Expert judgment refers to parameters or probabilities that have been assessed and elicited by domain experts. In this case, published data refer to data on human and technical reliability found in the literature and the accepted values for the aforementioned causation probability. Eleven models primarily use accident data to assess the risk level, which is collected from accident and incident databases and reports. Such models are not yet directly applicable for MASS, since they will be operated differently and rely on different technical solutions. Only six models use (discretized) real-time information to assess the current level of risk.

The next sections categorize the models, similar to the groups in Li et al. (2012), who reviewed ship risk models. The focus of the next sections is to generally describe the model types and assess their suitability for MASS generally.

5.1. Modelling categories

5.1.1. Models for assessing the risk in waterways

Collision and grounding risk models for waterways are often based on geometric models. The probability of an accident (P) is derived through the multiplication of two parameters, the probability to encounter a vessel that will result in a collision if no avoiding measures are taken (P_a) and the causation probability (P_C), which represents the probability that no evasive maneuver is taken (Fujii and Shiobara, 1971; MacDuff, 1974).

$$P = P_a \times P_c \tag{1}$$

The encounter probability is in most cases based on the geometrical traffic distribution in the fairway. The overlap between different fairways is used to find P_a for head-on collisions. For overtaking or crossing collisions similar considerations have been presented. A summary of possible methods for calculating the encounter probability can be found, for example, in Kristiansen (2005) or Li et al. (2012).

The grounding frequency can be determined similarly. For coastal areas or areas with shallow water, the ship traffic density can be determined and multiplied with a causation probability (Pedersen, 2010). This is based on the considerations of MacDuff (1974) and Fujii et al. (1974). One differentiates between powered groundings and drift groundings (Mazaheri et al., 2014).

The causation probability summarizes considerations of vessel maneuverability, crew, equipment, etc. (Pedersen, 2010). The probability is often determined through BBN, ETA, or FTA, or a combination of these. These methods will not be explained further. Both the encounter probability and causation probability may be derived from historical data on the traffic distribution in an area and the available accident data.

Models that fall in this category are M3-M5, M11, M16-M18, M26, M27, M30-M32, M34, M37, M41, M46, M48, M61 and M62. These models aim mostly at assessing the average risk in a waterway. They enable analysts to suggest regulatory measures for reducing the level of risk. Hence, these kind of models are not applicable to determine the level of risk of MASS, since MASS are not yet an integral part of the maritime traffic. In the future, these types of models need to account for MASS.

5.1.2. Causation probability models

Some publications present only a model for the causation probability once a vessel is on collision course. These models employ mostly BBN, ETA, and FTA. The models aim in many cases at one ship type, a specific fleet or a specific ship. Some address specific factors, such as, fatigue (M36), human operator performance (M12), or operation in arctic areas (M63).

Models that fall in this category are M7, M12, M15, M36, M38-M40, M42, M51, M52, M56, M58, M60, and M63. Where M7 and M15 are generic frameworks for risk modelling of maritime transport systems. These models may provide some basis for risk modelling of MASS, since they model certain risk aspects with a high level of detail. However, the focus of the models may not always be adequate.

5.1.3. Simulation approaches

To determine the encounter probability and consequently the accident risk, simulations may be used. These models frequently use a causation probability, which is derived through BBN, ETA, and FTA. However, not all models used for deriving the causation probability are presented by the literature.

The simulations use AIS data and other ship traffic data to simulate the paths of ships and identify potential collision candidates. Simulations may also be used to assess the allision risk or the grounding risk. The models are useful when areas with regular sea traffic shall be assessed, such as harbor areas, ferry or tanker traffic. Models that use simulations are M1, M6, M8, M13, M14, M19, M20, M22, M24 M25, M28, M29, M33, M35, M47, M49, M54, and M59.

Simulations, in general, may be useful to model the risk of MASS operation. Especially, for MASS being employed in route traffic it seems to be a promising tool. Characteristics of the MASS can be modeled and the behavior of the control software may be implemented. Particular traffic operating on the MASS route may be assessed and critical situations identified.

5.1.4. Real-time decision support

Several models and approaches have been developed to give real-time decision support to ship navigators and VTS operators. These approaches use underlying risk models in combination with calculation of the nearest point of approach to identify possible collision candidates. Models in this category are M9, M10, M21, M23, M44, M50, M55, and M61. These models may provide information to operators, however, they are not suitable for direct risk assessment for MASS. Such models do generally not model the ship in detail, since the focus lies on the surrounding vessels.

5.1.5. Other risk assessment approaches

Hu et al. (2007) (M2) used a fuzzy logic approach to the risk assessment of waterways. This may address uncertainties and probability ranges of scenarios. However, the model aims at specific waterways and hence their specific work has little relevance for MASS. Fuzzy logic, though, may be used to address the uncertainties in risk assessment of MASS.

Zaman et al. (2014) used a combination of FMEA and fuzzy logic to address the risk assessment of the strait of Malaga. They identify hazards for the strait and assess the magnitude of risk contribution. Hence, the knowledge gained from the model has few implications for MASS. However, the method may support the design of MASS.

Nivolianitou et al. (2016) presented a BBN for assessing the risk of ships passing an area. The assessment is based on accidents statistics using characteristics of vessels that have been involved in accidents. Such an approach is not suitable for risk assessment of MASS, since it is reactive and based on the accident statistics, which do not exist for MASS.

Wróbel et al. (2018) (M64) developed a STPA model to identify possible system hazards. The use of STPA reveals where control, through additional measures and functionalities is needed, to prevent the manifestation of hazards and consequently accidents. This model is further described in Section 6.

5.2. Parameters in the assessed ship risk models

This section provides an overview of parameters that have been used in the models. This corresponds to the second to last column in Table 3. This description forms the basis for the assessment of the models against the criteria outlined previously.

Each model considers several parameters that influence the probability of an accident. However, the number of parameters that are considered varies from model to model. Some models only consider a few vessel and fairway parameters, while others consider and describe in detail technical, human, environmental, and organizational factors that are considered. Thus, Table 3 contains a summary of parameters that have been included in the different models to give a comprehensive and comparable overview of the models. These parameters are used to assess the models against the identified criteria.

Traffic flow relates to the distribution of ship traffic over identified shipping lanes. The ship traffic is often Gaussian distributed. It contains information on the number of vessels passing a certain area, their trajectories and speed. Some models consider seasonal, daily, and hourly variations of the traffic flow. The traffic flow is often associated with the vessel traffic characteristics. These are the parameters of the vessels, such as ship type, length, width, and draught. Fairway characteristics refers to the dimensions of the waterway in question, in which the traffic is traveling. These are the length, width, and depth of the waterway and the spatial distribution of these. Several models split the fairway into several smaller segments to linearize meandering waterways. Geometric models make use of most of these parameters.

To be concise, environmental technical, human, and organizational factors that were similarly mentioned are presented in a summarized description in Table 3. For example, if human error was mentioned several times with respect to similar tasks (e.g., lookout), this is summarized as human error to avoid excessive repetition. Crew characteristics are used if several human and organizational factors were included (e.g., training, competence, experience, stress, alcohol consumption, tiredness, fatigue, etc.). With respect to environmental factors, weather describes the atmospheric environment. The sea state describes waves and currents with the associated directions. Visibility is mentioned as a separate factor, although dependent on weather. The reviewed models cover different levels of technical factors. Some models include failure of subsystems, (e.g., propulsion or navigational aid failure). Other models include very detailed failures (e.g., RADAR failure). Table 3 attempts to reflect these differences.

5.3. Evaluation against the criteria

Table 4 shows the results of the model evaluation against the criteria. Four models had insufficient information to assess all criteria. This is indicated in the table. Some models were assessed as partly fulfilling the criteria C1, C2, C5 C6, C7, and C8. This was the case in which models included considerations similar to the ones in the criteria. However, not enough information was presented to assure that these criteria are met.

Criterion 1 is fulfilled by 21 models, through failure of navigation aids. However, this is not a very detailed analysis of software systems. One model (M42) was assessed as partly fulfilling the criterion since technical reliability was mentioned as a factor. However, it was not clear if this referred also to hardware and software reliability.

To assess C2, the models were checked for human error and associated ergonomic considerations, such as navigational aid failure. 13 models fulfill criterion C2 and an additional 19 fulfill this criterion at least partly.

Twenty-four of the analyzed models fulfill C5 and include considerations for hardware, reliability, and maintenance. For C5, one model, M41 was assessed as partly meeting the criterion since only failure of the steering was mentioned.

C6 is addressed by six risk models. Three consider it partly, if the description of the events in the risk models indicated it, but did not explicitly model it. For the models fulfilling it, factors are included, such as, auxiliary systems.

Regarding C7, 14 models consider different operational modes. Most models consider different modes through the inclusion of pilotage or external assistance. Model M38 contains the autopilot as part of the considerations. Model M45 compares unmanned and conventional shipping and therefore includes different operational modes.

Only six models fulfil C3. Criterion 4 is fulfilled by 12 models. Ten models address C9. Ten models fulfilled six or more criteria. These are M7 (Trucco et al., 2008), M24 (Goerlandt et al., 2012; Goerlandt and Kujala, 2011; Hänninen and Kujala, 2010), M41 (Tvedt, 2014), M44 (Jensen, 2015), M45 (Khaled and Kawamura, 2015), M50 (Mazaheri et al., 2016), M51 (Haugen et al., 2016; Nilsen, 2016), M57, M58 (Hassel, 2017).

6. Discussion of the most promising models

Wróbel et al. (2018) (M64) used STPA to identify possible causes and contributors of the different system functions to system hazards. Almost all criteria, except for C4, which covers the interaction between operators are covered by Wróbel et al. (2018). The STPA method may be an important tool for the design and evaluation of MASS. STPA has also been used on for the assessment of dynamic positioning systems of ships to derive verification goals and identify hazards (Rokseth et al., 2016, 2017).

Wrobel et al. (2016) (M57) used a BBN to assess the risk level of MASS with respect to several possible accidents (collision, grounding, foundering, fire, or cargo related accidents). The BBN is divided into three levels. The first level represents the risk in relation to the aforementioned accidents. The second level summarizes possible initiating events, these are related to navigation, engineering, stability and buoyancy or miscellaneous. The third level summarizes causes to the accidents. Five main groups are identified; alerting, control algorithms, external information quality, maintenance regime, and sensors'

Table 4
Evaluation of the selected models against the criteria described in Table 1. Abbreviations: I. I. – insufficient information, N – No, P –Partly, Y – Yes.

Model	C1	C2	C3	C4	C5	C6	C7	C8	C9
M1	Y	P	N	N	Y	N	N	N	N
M2	N	N	N	N	N	N	N	N	N
M3	N	N	N	N	N	N	Y	N	N
M4	Y	P	N	N	Y	N	N	Y	N
M5	N	Y	N	N	Y	Y	N	N	N
M6	N	P	N	N	Y	N	N	Y	N
M7	Y	Y	N	Y	Y	N	Y	Y	Y
M8	N	N	N	N	N	N	N	N	N
M9	N	N	N	N	N	N	N	N	N
M10	N	N	N	N	N	N	N	N	N
M11	N	P	N	N	N	N	N	N	N
M12	N	N	N	Y	N	N	N	Y	N
M13	Y	P	N	N	Y	N	Y	Y	N
M14	N	P	N	N	Y	N	Y	Y	N
M15	N	N	N	N	N	N	N	N	N
M16	Y	P	N	N	Y	N	N	Y	N
M17	N	N	N	N	N	N	P	N	N
M18	N	P	N	N	N	N	N	N	N
M19	N	N	N	N	N	N	N	N	N
M20	I. I.	P	I. I.	I. I.	Y	I. I.	I. I.	I. I.	I. I.
M21	N	N	N	N	N	N	N	N	N
M22	Y	P	N	N	Y	N	N	N	N
M23	N	N	N	N	N	N	N	N	N
M24	N	Y	N	N	N	N	Y	Y	Y
M25	Y	Y	N	Y	Y	N	Y	Y	N
M26	N	N	N	N	N	N	N	N	N
M27	N	N	N	N	N	N	N	N	N
M28	N	P	N	N	Y	N	N	N	N
M29	N	N	N	N	N	N	Y	N	N
M30	N	P	N	N	Y	N	N	N	N
M31	N	N	N	N	N	N	N	N	N
M32	N	N	N	N	N	N	N	N	N
M33	I. I.	I. I.	I. I.	I. I.	I. I.	I. I.	I. I.	I. I.	I. I.
M34	N	N	N	N	N	N	N	N	N
M35	N	N	N	N	N	N	N	N	N
M36	N	Y	Y	Y	N	N	N	Y	Y
M37	N	N	N	N	N	N	N	N	N
M38	Y	N	N	N	Y	Y	N	Y	N
M39	Y	P	N	N	Y	N	N	N	N
M40	Y	Y	N	N	N	N	N	N	N
M41	Y	P	N	N	P	N	Y	Y	N
M42	P	Y	N	Y	Y	Y	Y	Y	Y
M43	N	P	N	N	Y	N	N	N	N
M44	N	N	N	N	N	N	N	N	N
M45	Y	P	Y	N	Y	Y	Y	Y	Y
M46	Y	P	N	Y	Y	N	Y	Y	Y
M47	N	N	N	N	N	N	N	N	N
M48	N	N	N	N	N	N	N	N	N
M49	N	N	N	N	N	N	N	N	N
M50	N	N	N	N	N	N	N	N	N
M51	N	Y	N	Y	Y	Y	Y	Y	Y
M52	Y	Y	N	Y	Y	P	N	Y	Y
M53	N	N	N	N	N	N	N	N	N
M54	N	N	N	N	N	N	N	N	N
M55	Y	N	N	Y	N	N	N	Y	N
M56	Y	Y	N	Y	N	N	N	Y	N
M57	Y	Y	Y	N	Y	I.I.	Y	P	N
M58	Y	P	N	Y	N	N	N	P	N
M59	I. I.	I. I.	I. I.	I. I.	I. I.	I. I.	I. I.	I. I.	N
M60	Y	Y	Y	Y	Y	P	N	Y	Y
M61	N	N	N	N	N	N	N	N	N
M62	N	N	N	N	N	N	N	N	N
M63	Y	Y	Y	N	N	P	N	N	N
M64	Y	P	Y	N	Y	Y	Y	Y	Y

performance. The groups and their possible inclusion are not further described or developed, and the model is not quantified.

Wrobel et al. (2016) address several important issues with their model. Therefore, it may form a suitable basis for further development. However, assessing several accident types in one model, may be a major challenge, since a variety of risk influencing factors may interact in different ways for different accidents.

Jensen (2015) presented a risk assessment (M44) for a prototype unmanned bulk carrier using ETA and FTA, following the FSA process. In addition to ship-ship collisions, foundering of the vessel is investigated. The models are used to compare conventional with autonomous operation. Therefore, the models have been specifically developed for autonomous ships. Communication between the members of the SCC crew is not included, and human factors are only considered for the manned case. However, human factors should be included in a revised version of the model for remote control.

The collision encounter probability is assessed with geometric models based on the International Association of Marine Aids to Navigation and Lighthouse Authorities (IALA) Waterways Risk Management Program (IWRAP) Mark 2. This may be a good starting point to assess the possible encounters on a long voyage route. However, simulations to assess the possible encounters may be more suitable in areas where traffic patterns vary strongly during a day or for traffic on a specific route (Li et al., 2012).

Overall, the models in M45 outline well how a risk analysis may be structured using FTA and ETA. The presented models include high-level function failures of the equipment, engine, steering, software, and hardware. For a real system, these function failures need to be modeled in more detail to represent the ship and its particulars. The models lack detail in terms of the control system components. However, these are essential parts of a MASS and need to be considered.

Trucco et al. (2008) presented a general framework (M7) for risk assessment of maritime transportation systems. A BBN is used to model the interaction of human, organizational, and technical factors, which influence the basic event probability of fault trees. The fault trees are used to assess the probability of accidental events. As a case study, Trucco et al. (2008) assessed the collision probability of a high-speed vessel. They consider three elements leading to a collision: human errors, automation and mechanical failures, and maneuvering errors.

The modelling framework developed by Trucco et al. (2008) seems appropriate as a starting point for the development of risk models for MASS. The interaction between different risk influencing factors is an important contributor to the level of risk and may be captured through BBN. The accidental chain of events can be modelled through FTA. Hence, such a framework, together with the framework by Vanem et al. (2009), could be considered as basis for the development of the risk models.

Tvedt (2014) presented a risk assessment framework (M42) for allision scenarios between an offshore supply vessel and an offshore platform. Three scenarios were identified. Tvedt (2014) used ETA to model the chain of events in the identified scenarios. Failures of mitigating barriers are modeled with FTA. The basic events in the FTAs are assessed by BBNs, including human and organizational factors that influence the level of risk. These factors are identified from different sources and include a wide range of considerations, such as HMI usability, training, communication, personal factors of the crew, maintenance, reliability, and manning. The model is not quantitative and is limited to an offshore supply vessel approaching an offshore platform. Similar methods are used as suggested by Trucco et al. (2008) and especially the operator model seems promising to transfer to a risk model for MASS. The model itself, due to its focus on offshore platforms cannot be transferred to the case of MASS. However, the models in the scenarios may need adaptation to account for MASS in the future.

Mazaheri et al. (2016) developed a BBN (M51) for the assessment of grounding probability of a marine traffic system, such as a vessel, vessel type, or a certain waterway. Mazaheri et al. (2016) based their model on incident and accident reports and earlier models. This makes the model generally unsuitable. In addition, the model does not provide further guidance on how the factors in the BBN may be assessed with respect to their not available data or for other ship systems.

Goerlandt et al. (2012), Goerlandt and Kujala (2011), and Hänninen and Kujala (2010) (M25) presented models to assess the risk associated with tanker collisions in a waterway; hence, it treats the ship parameters rather superficially and is only limitedly suitable to assess the risk level of the ship. Goerlandt et al. (2012) presented the overall methodology for risk assessment, using simulation including ship particulars, route information, departure time, and speed, following Goerlandt and Kujala (2011) and Hänninen and Kujala (2010) for the assessment of the collision frequency. Goerlandt and Kujala (2011) assessed the encounter frequency of vessels in a specific waterway. Hänninen and Kujala (2010) presented the model for assessing the causation probability. The causation probability represents evasive maneuvers by the two vessels and is assessed through a BBN. Hänninen and Kujala (2010) included several technical, human, environmental, and organizational factors. The approach may be further developed or used as guideline to assess the risk level of MASS operating in waterways.

Hassel (2017) assessed the allision risk (M60) for offshore oil and gas platforms. The BNN model focuses on both aspects related to the platform and the ship on collision course. Similar to M42, M60 addresses the allision risk from the perspective of the offshore platform. Hence, the model may need to be adapted, to assess the change of the allision risk level of offshore platforms by MASS. All aspects of communication (C3, C4, and C8) are covered. Model 60 may be used as guideline, how these aspects can be included in a BBN model for MASS.

Khaled and Kawamura (2015) assessed the collision risk (M46) in a harbor area. They used the geometric model implemented in IWRAP (Friis-Hansen, 2008) to assess an encounter frequency and combine it with an adapted BBN to assess the causation probability. The BBN includes, among others, environmental factors, personal factors of crew members, human error, and technical reliability of navigational equipment and communication equipment. Khaled and Kawamura (2015) included considerations that are relevant for operation of MASSs. However, they are covered only superficially since the model was made for risk assessment of waterways. Since the model is designed for harbor areas, it may provide input for assessing the risk level of MASS when approaching ports.

Model 52 is the Norwegian national ship risk model (Nilsen (2016); Haugen et al. (2016)). The model was developed for the risk assessment and implementation of risk reduction measures in Norwegian waterways. The model does not consider different operational modes and communication between vessel operators. Only the detailed model for groundings is available; hence, these considerations might be included in a collision model. Since the model focuses on waterways and is based on historical data for incidents and accidents, it is not suited to demonstrate safety compliance of MASS. The model and work around the model include different ship types and their risk levels MASS may be included in the future.

In summary, the literature provides some suggestions for the conduction of risk assessments for MASS. The STPA methods seems to be suitable tool for analyzing possible hazards and proposing risk reduction measures.

Some of the analyzed models focus on specific waterways and locations. The different foci result in various aspects that are included and highlighted in the models. To demonstrate a sufficiently low-risk level of an MASS, it is necessary to model its behavior and particulars in detail, which may require risk modelling from different risk perspectives on the MASS.

In some cases, a quantitative assessment is necessary, to show that the risk level has been addressed by suitable measures. Models that are used currently for the risk assessment of conventional ships, may provide insight into how a model could be developed. Building risk models of MASS may find a starting point in risk models for conventional ships. However, the risk influencing factors in the models and their quantification need to be elicited for the MASS case.

Areas that need special attention, for example, software, and remote control and associated human operator considerations, are rarely covered in depth. Different approaches are needed to include these considerations in risk models for MASS.

7. Conclusion

This article reviews current risk models for ship collisions and groundings, which have been presented in the literature since 2005. The 64 analyzed models mainly aim at assessing the ship collision frequency, grounding frequency, or frequency of allisions in a certain waterway or geographical area. Most models use a geometrical modeling approach, often in combination with other modeling techniques, to determine the frequencies or probabilities of the accident. Models aiming at risk assessment of a waterway treat ships superficially with respect to relevant factors, such as technical equipment and its reliability. Hence, such models are not applicable to demonstrate the risk level of a ship.

Nine criteria are used to assess the identified relevant risk models with respect to their applicability to MASS operation. A systems engineering approach was used to identify the criteria. The criteria cover relevant aspects for the operation of MASS: component and subsystem redundancy, different operational modes, HMI, communication among different involved actors, technical reliability, maintenance, software reliability and manning. These criteria cover a broad range of aspects since the current concepts for MASS vary among each other, which does not allow for a more detailed system evaluation.

Ten models fulfill six or more criteria. These were investigated more closely. Seven models that were closely investigated in this article are based on conventional ship operation. The operation of MASS will be different from conventional ship operation. Technical reliability, software reliability, and the situation awareness of the operators become even more important in MASS. The models developed for MASS address most relevant issues. However, due to the lack of certainty on design and operational concepts, these models are rather superficial. No models can be defined without concrete operational concepts and clear system definitions which makes an in-depth analysis and assessment of the reviewed models difficult.

The evaluation presented in this article shows that some of the current conventional ship risk models and the underlying frameworks could be used as a starting point for developing risk models for MASS. The structure and considerations included in the models should be further considered regarding risk modeling of MASS.

The quantification of ship risk models traditionally is based on accident and incident data, but such an approach is not yet applicable for risk models of MASS. Hence, expert assessments and test data need to be derived and used if a quantified risk assessment is attempted.

One issue that all the analyzed models have in common (except for M57 (Wrobel et al., 2016) and M64 (Wróbel et al., 2018)), is that they do not include the communication connection with a shore base. This is one of the main requirements for MASS, that they can be remotely controlled and supervised. Even if MASS have minimal crew on board, part of the vessel will be highly automated, and situation assessment requires a robust communication line between the vessel and competent personnel on shore.

Seven of the ten models discussed in more detail have one aspect in common; they use BBN for at least as part of the risk model. Only Jensen (2015) and Wróbel et al. (2018) do not use a BBN. Hänninen (2014) highlighted the usability and usefulness of BBNs for maritime safety management. With the flexibility of the modeling method and the input from experts, it is possible to build risk models for MASS operation. Hence, BBNs should be considered part of a risk model for MASS operation. A systems engineering approach might benefit the development of such a risk model in identifying comprehensive system requirements.

A dedicated MASS risk model should focus on the assessment of the control and software system and the effects of its failure. Current models do not consider this aspect. Dedicated methods for assessment of software failure and control systems need to be applied. Currently used modeling techniques in the ship risk models are not sufficient since software behaves deterministically (Chu et al., 2009). Methods that may be used could be, among others, STPA (Leveson et al., 2012), or the Functional Resonance Analysis Method (Hollnagel, 2012), which has been already employed in accident investigation of maritime accidents (Tian et al., 2016).

Other aspects that need more attention in the future are the interactions between conventional and autonomous ships since MASS will not replace all maritime vessels in the foreseeable future. Further investigation should include the effects of MASS on traffic patterns. The methods relating to the geometrical analysis of collision frequency might need to be adapted to new traffic patterns. In addition, permanent navigational aids along the coast and in waterways may need to be changed to facilitate navigation of MASS. Current aids, such as navigational lights and buoys, assist the human navigators using RADAR or similar equipment with visual perception for verification. This is also an area that needs to be further investigated and that may affect the risk related to MASS.

Financial support

I. B. Utne and C. A. Thieme appreciate the support of the Research Council of Norway through the Centres of Excellence funding scheme, Project number 223254 – NTNU AMOS.

Acknowledgments

The authors want to thank the two anonymous reviewers, who made valuable comments on an earlier version of this article and helped to improve it.

References

Advanced Autonomous Waterborne Applications, 2016. Remote and autonomous ships - the next steps. In: Laurinen, M. (Ed.), Adavanced Autonomous Waterborne Applications London, pp. 88 UK.

Afenyo, M., Khan, F., Veitch, B., Yang, M., 2017. Arctic shipping accident scenario analysis using Bayesian Network approach. Ocean Eng. 133, 224–230.

Akhtar, M.J., Utne, I.B., 2014. Human fatigue's effect on the risk of maritime groundings - a Bayesian Network modeling approach. Saf. Sci. 62 (0), 427–440.

Almaz, O.A., 2012. Risk and Performance Analysis of Ports and Waterways - the Case of Delaware River and Bay. Graduate School-New Brunswick. Rutgers, The State University of New Jersey, New Brunswick, New Jersey.

Aven, T., Zio, E., 2014. Foundational issues in risk assessment and risk management. Risk Anal. 34 (7), 1164–1172.

Bertram, V., 2016. Autonomous ship technology -smart for sure, unmanned maybe. In: Griffiths, M. (Ed.), Smart Ship Technology. The Royal Institute of Naval Architects, London, UK, pp. 5–112.

Blanchard, B.S., 2008. System Engineering Management, fourth ed. Wiley, Hoboken, N.J ed.

Blokus-Roszkowska, A., Smolarek, L., 2013. Application of simulation methods for evaluating The sea waterways traffic organisation. ISRN Applied Mathematics 2013, 1–8.

Brito, M.P., Griffiths, G., 2016. A Bayesian approach for predicting risk of autonomous underwater vehicle loss during their missions. Reliab. Eng. Syst. Saf. 146, 55–67.

Brito, M.P., Griffiths, G., Challenor, P., 2010. Risk analysis for autonomous underwater vehicle operations in extreme environments. Risk Anal. 30 (12), 1771–1788.

Brito, M.P., Griffiths, G., Ferguson, J., Hopkin, D., Mills, R., Pederson, R., MacNeil, E., 2012. A behavioral probabilistic risk assessment framework for managing autonomous underwater vehicle deployments. J. Atmos. Ocean. Technol. 29 (11), 1689–1703.

Burmeister, H.C., Walther, L., Jahn, C., Töter, S., Froese, J., 2014. Assessing the frequency and material consequences of collisions with vessels lying at an anchorage in line with IALA iWrap MkII. TransNav, the International Journal on Marine Navigation and Safety of Sea Transportation 8 (1), 61–68.

Chai, T., Weng, J., De-qi, X., 2017. Development of a quantitative risk assessment model for ship collisions in fairways. Saf. Sci. 91, 71–83.

Chin, H.C., Debnath, A.K., 2009. Modeling perceived collision risk in port water navigation. Saf. Sci. 47 (10), 1410–1416.

Chu, T.-L., Martinez-Guridi, G., Yue, M., Samanta, P., Vinod, G., Lehner, J., 2009. Workshop on philosophical basis for incorporating software failures in a probabilistic risk assessment, digital system software PRA. Brookhaven Natl. Lab. 188.

Copping, A., Breithaupt, S., Whiting, J., Grear, M., Tagestad, J., Shelton, G., 2016. Likelihood of a marine vessel accident from wind energy development in the Atlantic. Wind Energy 19 (9), 1557–1566.

COWI, 2008. Risk Analysis of Sea Traffic in The Area Around Bornholm, Kongens Lyngby, Denmark. pp. 150.

COWI, 2012. Project on Sub-Regional Risk of Spill of Oil and Hazardous Substances in The Baltic Sea (BRISK) - Risk Method Note, Kongens Lyngby, Denmark.

Curley, R., 2012. Chapter 4: ship operation. In: Levy, M.I. (Ed.), Transportation and Society: Complete History of Ships and Boats: from Sails and Oars to Nuclear-powered Vessels (1). Britannica Educational Publishing, New York, NY, pp. 53–74.

Danish Maritime Authority, 2018. Analysis of regulatory barriers to the use of autonomous ships, regulatory scoping exercise for the use of maritime autonomous surface ships (MASS). In: Maritime Safety Committee, Danish Maritime Authority (DMA), Danmark, pp. 143.

Debnath, A.K., 2009. Traffic-conflict-based Modeling of Collision Risk in Port Waters, Dept. Of Civil Engineering. National University of Singapore, Singapore, pp. 153.

Debnath, A.K., Chin, H.C., 2010. Navigational traffic Conflict technique: a proactive approach to quantitative measurement of collision risks in port waters. J. Navig. 63 (01), 137.

DNV-GL, 2015. The ReVolt. DNV GL AS.

Ellis, J., Forsman, B., Hüffmeier, J., Johansson, J., 2008. Methodology for Assessing Risks to Ship Traffic from Offshore Wind Farms, VINDPILOT-report. SSPA Sweden AB, Gøteborg, Sweden. pp. 152.

Endsley, M.R., Kaber, D.B., 1999. Level of automation effects on performance, situation awareness and workload in a dynamic control task. Ergonomics 42 (3), 462–492.

Federation, Nautilus, 2018. Report of a Survey on what Maritime Professionals Think about Autonomous Shipping, Regulatory Scoping Exercise for the Use of Maritime Autonomous Surface Ships (MASS). International Maritime Organization, Maritime Safety Committee, London, UK.

Friis-Hansen, P., 2008. IWRAP MK II - Basic Modelling Principles for Prediction of Collision and Grounding Frequencies, IWRAP MK II, Rev, 4 ed. .

Fujii, Y., Shiobara, R., 1971. The analysis of traffic accidents. J. Inst. Navig. 24 (4), 534–543.

Fujii, Y., Yamanouchi, H., Mizuki, N., 1974. Some factors affecting the frequency of accidents in marine traffic ii - the probability of stranding. J. Inst. Navig 27 (2), 239–243.

Goerlandt, F., Kujala, P., 2011. Traffic simulation based ship collision probability modeling. Reliab. Eng. Syst. Saf. 96 (1), 91–107.

Goerlandt, F., Montewka, J., 2015. Maritime transportation risk analysis: review and analysis in light of some foundational issues. Reliab. Eng. Syst. Saf. 138, 115–134.

Goerlandt, F., Hänninen, M., Ståhlberg, K., Montewka, J., Kujala, P., 2012. Simplified risk analysis of tanker collisions in the Gulf of Finland. TransNav, the International Journal on Marine Navigation and Safety of Sea Transportation 6 (3), 381–387.

Goerlandt, F., Montewka, J., Kuzmin, V., Kujala, P., 2015. A risk-informed ship collision alert system: framework and application. Saf. Sci. 77, 182–204.

Griffiths, G., Brito, M.P., 2008. Predicting risk in missions under sea ice with autonomous underwater vehicles. Autonomous Underwater Vehicles 2008, 1–7 AUV 2008. IEEE/OES.

Hänninen, M., 2014. Bayesian networks for maritime traffic accident prevention: benefits and challenges. Accid. Anal. Prev. 73, 305–312.

Hänninen, M., Kujala, P., 2010. The effects of causation probability on the ship collision statistics in the Gulf of Finland. TransNav, the International Journal on Marine Navigation and Safety of Sea Transportation 4 (1), 79–84.

Hassel, M., 2017. Risk Analysis and Modelling of Allisions between Passing Vessels and Offshore Installations, Dept. Marine Technology. Norwegian University of Science and Technology (NTNU), Trondheim, Norway, pp. 82.

Hassel, M., Utne, I.B., Vinnem, J.E., 2014. Analysis of the Main Challenges with the Current Risk Model for Collisions between Ships and Offshore Installations on the Norwegian Continental Shelf, PSAM 2014-Probabilistic Safety Assessment and Management.

Hassel, M., Utne, I.B., Vinnem, J.E., 2017. Allision risk analysis of offshore petroleum installations on the Norwegian Continental Shelf—an empirical study of vessel traffic patterns. WMU Journal of Maritime Affairs 16 (2), 175–195.

Haugen, S., Almklov, P.G., Nilsen, M., Bye, R.J., 2016. Norwegian national Ship Risk Model, Maritime Technology and Engineering III. CRC Press, pp. 831–838.

Hollnagel, E., 2012. FRAM – the Functional Resonance Analysis Method, 1. Edition. Ashgate, Farnham, UK.

Hu, S., Fang, Q., Xia, H., Xi, Y., 2007. Formal safety assessment based on relative risks model in ship navigation. Reliab. Eng. Syst. Saf. 92 (3), 369–377.

Huang, Y., Van Gelder, P.H.A.J.M., Mendel, M.B., 2017. Imminent ships collision risk assessment based on velocity obstacle. In: Walls, L., Revie, M., Bedford, T. (Eds.), 26th European Safety and Reliability Conference, ESREL 2016. CRC Press/Balkema, pp. 111.

IMO, 2002. Guidelines for Formal safety Assessment (FSA) for Use in The Imo Rule-Making Process. International Maritime Organization, London, UK.

Insaurralde, C.C., 2012. Autonomic Management for the Next Generation of Autonomous Underwater Vehicles, 2012 IEEE/OES Autonomous Underwater Vehicles, AUV 2012. IEEE, Southampton; United Kingdom, pp. 1–8.

Jensen, F., 2015. Hazard and Risk Assessment of Unmanned Dry Bulk Carriers on the High Seas, Dept. Of Naval Architecture. Technische Universität Hamburg Harburg, Hamburg, pp. 157.

Jeong, J.S., Park, G.K., Kim, K.I., 2012. Risk assessment model of maritime traffic in time-variant CPA environments inwaterway. J. Adv. Comput. Intell. Intell. Inf. 16 (7), 866–873.

Johansson, J., Molitor, E., 2011. Risk Assessment of the Vessel Traffic in the Kattegat Including Effects of Traffic Separation Schemes from the Skaw to the Sound – Oil Spill Accidents Relevant for the Coast of Halland, 1 ed. (SSPA Sweden AB, Göteborg, Sweden).

Kaneko, F., 2010. A method for estimation of grounding frequency by using trajectories of ships and geometry of seabed. In: Ehlers, S., Romanoff, J. (Eds.), 5th International Conference on Collision and Grounding of Ships. Multiprint Oy, Espoo, Finland, pp. 123–132.

Kaneko, F., 2012. Models for estimating grounding frequency by using ship trajectories and seabed geometry. Ships Offshore Struct. 7 (1), 87–99.

Khaled, M.E., Kawamura, Y., 2015. Collision risk analysis of Chittagong port in Bangladesh by using collision frequency calculation models with modified BBN model. In: Proceedings of the Twenty-fifth (2015) International Ocean and Polar Engineering Conference. ISOPE, Kona, Big Island, Hawaii, USA, pp. 829.

Khan, F.I., Yang, M., Veitch, B., Ehlers, S., Chai, S., 2014. Transportation risk analysis framework for Arctic waters. In: Proceedings of the International Conference on Offshore Mechanics and Arctic Engineering - OMAE.

Khan, B., Khan, F.I., Veitch, B., Yang, M., 2018. An operational risk analysis tool to analyze marine transportation in Arctic waters. Reliab. Eng. Syst. Saf. 169, 485–502.

Klemola, E., Kuronen, J., Kalli, J., Arola, T., Hänninen, M., Lehikoinen, A., Kuikka, S., Kujala, P., Tapaninen, U., 2009. A cross-disciplinary approach to minimising the risks of maritime transport in the Gulf of Finland. World Rev. Intermodal Transp. Res. 2 (4), 343.

Kretschmann, L., Rødseth, Ø.J., Sage Fuller, B., Noble, H., Horahan, J., McDowell, H., 2015a. D9.3: Quantitative Assessment, Maritime Unmanned Navigation through Intelligence in Networks.

Kretschmann, L., Rødseth, Ø.J., Tjora, Å., Sage Fuller, B., Noble, H., Horahan, J., 2015b. D9.2: Qualitative Assessment, Maritime Unmanned Navigation through Intelligence in Networks, 1.0 ed. pp. 45.

Kristiansen, S., 2005. Maritime Transportation - Safety Management and Risk Analysis. Elsevier Butterworth-Heinemann, Oxford, Burlington.

Leveson, N.G., 2011. Engineering a Safer World - System Thinking Applied to Safety. The MIT Press, Cambridge, Massachusetts, USA; London, England.

Leveson, N.G., Fleming, C.H., Spencer, M., Thomas, J., Wilkinson, C., 2012. Safety assessment of Complex, software-intensive systems. SAE International Journal of Aerospace 5 (1), 233–244.

Li, S., Meng, Q., Qu, X., 2012. An overview of maritime waterway quantitative risk assessment models. Risk Anal. 32 (3), 496–512.

Lim, G.J., Cho, J., Bora, S., Biobaku, T., Parsaei, H., 2018. Models and computational algorithms for maritime risk analysis: a review. Ann. Oper. Res.

Ma, F., Chen, Y.W., Huang, Z.C., Yan, X.P., Wang, J., 2016. A novel approach of collision assessment for coastal radar surveillance. Reliab. Eng. Syst. Saf. 155, 179–195.

MacDuff, T., 1974. The probability of vessel collision. Ocean Ind. 9 (9), 144–148.

Maritime, Kongsberg, 2017. YARA and KONGSBERG Enter into Partnership to Build World's first Autonomous and Zero Emissions Ship. Kongsberg. (online article).

Martins, M.R., Maturana, M.C., 2009. The application of the bayesian networks in the human reliability analysis. In: ASME 2009 International mechanical Engineering Congress and Exposition. Asme, Lake Buena Vista, Florida, USA, pp. 341–348.

Mazaheri, A., Montewka, J., Kujala, P., 2014. Modeling the risk of ship grounding—a literature review from a risk management perspective. WMU Journal of Maritime Affairs 13 (2), 269–297.

Mazaheri, A., Montewka, J., Kujala, P., 2016. Towards an evidence-based probabilistic risk model for ship-grounding accidents. Saf. Sci. 86, 195–210.

Merrick, J.R., van Dorp, R., 2006. Speaking the truth in maritime risk assessment. Risk Anal. 26 (1), 223–237.

Montewka, J., Hinz, T., Kujala, P., Matusiak, J., 2010. Probability modelling of vessel collisions. Reliab. Eng. Syst. Saf. 95 (5), 573–589.

Montewka, J., Krata, P., Goerlandt, F., Mazaheri, A., Kujala, P., 2011. Marine traffic risk modelling - an innovative approach and a case study. Proc. Inst. Mech. Eng. O J. Risk Reliab. 225 (O3), 307–322.

Montewka, J., Ehlers, S., Tabri, K., 2012a. Modelling risk of a collision between a LNG tanker and a harbour tug. Marine Systens & Ocean Technology 7 (1), 3–13.

Montewka, J., Goerlandt, F., Kujala, P., 2012b. Determination of collision criteria and causation factors appropriate to a model for estimating the probability of maritime accidents. Ocean Eng. 40, 50–61.

Montewka, J., Goerlandt, F., Innes-Jones, G., Owen, D., Hifi, Y., Porthin, M., 2014. Quantifying the Effect of Noise, Vibration and Motion on Human Performance in Ship Collision and Grounding Risk Assessment. PSAM 2014-Probabilistic Safety Assessment and Management.

Mulyadi, Y., Kobayashi, E., Wakabayashi, N., Pitana, T., Wahyudi, 2014. Development of ship sinking frequency model over Subsea Pipeline for Madura Strait using AIS data. WMU Journal of Maritime Affairs 13 (1), 43–59.

MUNIN, 2012. Maritime Unmanned Navigation through Intelligence in Networks.

Nilsen, M., 2016. National Ship Risk model and the human factors - perspectibe on investigation report analysis. National Ship risk model 79.

Nivolianitou, Z.S., Koromila, I.A., Giannakopoulos, T., 2016. Bayesian network to predict environmental risk of a possible ship accident. Int. J. Risk Assess. Manag. 19 (3), 228–239.

Norwegian Maritime Authority, 2016. World's First Test Area for Autonomous Ships Opened. Sjøfartsdirektoratet, Haugesund, Norway.

Norwegian Shipowners' Association, 2003. Part I - General Part, Superintendent's Manual. Norwegian Shipowners' Association, Oslo.

Ozbas, B., Or, İ., Uluscu, O.S., Altiok, T., 2009. Simulation-based risk analysis of maritime transit traffic in the strait of istanbul. TransNav, the International Journal on Marine Navigation and Safety of Sea Transportation 3 (3), 295–300.

Parasuraman, R., Sheridan, T.B., Wickens, C.D., 2000. A model for types and levels of human interaction with automation. IEEE Trans. Syst. Man Cybern. Syst. Hum. 30 (3), 286–297.

Pedersen, P.T., 2010. Review and application of ship collision and grounding analysis procedures. Mar. Struct. 23 (3), 241–262.

Povel, D., Bertram, V., Steck, M., 2010. Collision Risk Analyses for Offshore Wind Energy Installations. Proceedings of the International Offshore and Polar Engineering Conference, pp. 745–751.

Presencia, C.E., Shafiee, M., 2017. Risk analysis of maintenance ship collisions with offshore wind turbines. Int. J. Sustain. Energy 1–21.

Przywarty, M., 2008. Models of ships Groudings on coastals areas. J. Konbin 5 (2).

Przywarty, M., Gucma, L., Marcjan, K., Bak, A., 2015. Risk analysis of collision between passenger ferry and Chemical tanker in the western zone of the Baltic Sea. Pol. Marit. Res. 22 (2), 3–8.

Rasmussen, F.M., Glibbery, K.A.K., Melchild, K., Hansen, M.G., Jensen, T.K., Lehn-Schiøler, T., Randrup-Thomsen, S., 2012. Quantitative assessment of risk to ship traffic in the fehmarnbelt fixed ink project. Journal of Polish Safety and Reliability Association 3 (No. 1), 123–134.

Rausand, M., 2011. Risk Assessment - Theory, Methods, and Applications, first ed. John Wiley & Sons, Hoboken, New Jersey, USA.

Rekha, A.G., Ponnambalam, L., Abdulla, M.S., 2016. Predicting maritime groundings using support vector data description model. Communications in Computer and Information Science 575, 329–334.

Ren, Y., Mou, J., Yan, Q., Zhang, F., 2011. Study on assessing dynamic risk of ship collision, ICTIS 2011: multimodal approach to sustained transportation system development - information, technology, implementation. In: Proceedings of the 1st Int. Conf. On Transportation Information and Safety, pp. 2751–2757.

Rødseth, Ø.J., Tjora, Å., Baltzersen, P., 2014. D4.5 Architecture Specification, Maritime Unmanned Navigation through Intelligence in Networks.

Rødseth, Ø.J., Burmeister, H.-C., 2015a. D10.2-New ship designs for autonomous vessels. In: Rødseth, Ø.J. (Ed.), MUNIN-FP7 GA-No 314286.

Rødseth, Ø.J., Burmeister, H.-C., 2015b. Risk assessment for an unmanned merchant ship. TransNav, the International Journal on Marine Navigation and Safety of Sea Transportation 9 (3), 357–364.

Rødseth, Ø.J., Nordahl, H., 2017. Definitions for Autonomous Merchant Ships. NFAS - Norwegian Forum for Autonomous Ships, Trondheim, Norway.

Rødseth, Ø.J., Tjora, Å., 2014. A Risk Based Approach to the Design of Unmanned Ship Control Systems, Maritime-port Technology and Development. CRC Press, pp. 153–161.

Rokseth, B., Utne, I.B., Vinnem, J.E., 2016. A systems approach to risk analysis of maritime operations. Proceedings of the Institution of Mechanical Engineers, Part O. J. Risk Reliab 1748006X16682606.

Rokseth, B., Utne, I.B., Vinnem, J.E., 2017. Deriving Verification Objectives and Scenarios for Maritime Systems Using the Systems-theoretic Process Analysis. Reliability Engineering & System Safety.

Senol, Y.E., Sahin, B., 2016. A novel real-time Continuous fuzzy fault tree analysis (RC-FFTA) model for dynamic environment. Ocean Eng. 127, 70–81.

Sheridan, T.B., 2011. Adaptive automation, level of automation, allocation authority, supervisory control, and adaptive control: distinctions and modes of adaptation. IEEE transactions on systems, man and cybernetics. Part A (Systems and Humans) 41 (4), 662–667.

Sheridan, T.B., Verplank, W.L., 1978. Human and Computer Control of Undersea Teleoperators. DTIC Document.

Silveira, P.A.M., Teixeira, A.P., Soares, C.G., 2013. Use of AIS data to Characterise marine traffic patterns and ship collision risk off the coast of Portugal. J. Navig. 66 (6), 879–898.

SN-ISO, 2010. SN-ISO Guide 73:2009 Risk Management Terminology. International Organization for Standardization, Lysaker, Norway, pp. 17.

Sotiralis, P., Ventikos, N.P., Hamann, R., Golyshev, P., Teixeira, A.P., 2016. Incorporation of human factors into ship collision risk models focusing on human centred design aspects. Reliab. Eng. Syst. Saf. 156, 210–227.

Suman, S., Nagarajan, V., Sha, O.P., Khanfir, S., Kobayashi, E., Malik, A.M.b.A., 2012. Ship Collision Risk Assessment Using AIS Data.

Thieme, C.A., Utne, I.B., 2017. A risk model for autonomous marine systems and operation focusing on human–autonomy collaboration. Proceedings of the Institution of Mechanical Engineers, Part O. J. Risk Reliab. 231 (4), 446–464.

Thieme, C.A., Utne, I.B., Schjølberg, I., 2015b. Risk modeling of autonomous underwater vehicle operation focusing on the human operator. In: Podofillini, L., Sudret, B., Stojadinovic, B., Zio, E., Kröger, W. (Eds.), 25th European Safety and Reliability Conference, ESREL 2015. CRC Press, Taylor & Francis Group, Zürich, Switzerland, pp. 3653–3660.

Thieme, C.A., Utne, I.B., Schjølberg, I., 2015a. A Risk Management Framework for Unmanned Underwater Vehicles Focusing on Human and Organizational Factors Proceedings of the ASME 2015 34th International Conference on Ocean. Offshore and Arctic Engineering OMAE2015. ASME, St. John's, NL, Canada.

Tian, J., Wu, J., Yang, Q., Zhao, T., 2016. FRAMA: a safety assessment approach based on Functional Resonance Analysis Method. Saf. Sci. 85, 41–52.

Trucco, P., Cagno, E., Ruggeri, F., Grande, O., 2008. A Bayesian Belief Network modelling of organisational factors in risk analysis: a case study in maritime transportation. Reliab. Eng. Syst. Saf. 93 (6), 845–856.

Tvedt, F.E., 2014. Risk Modelling of Collision between Supply Ships and Oil- and Gas Installations, Dept. Of Production and Quality Engineering. Norwegian University of Science and Technology, Trondheim, Norway, pp. 97.

Tvete, H.A., 2015. ReVolt; the Unmanned, Zero Emission, Short Sea Ship of Hte Future, Green Ship Technology 2015, Copenhagen, Danmark.

Uluscu, O.S., Ozbas, B., Altiok, T., Or, I., 2009. Risk analysis of the vessel traffic in the strait of istanbul. Risk Anal. 29 (10), 1454–1472.

UNCLOS, 1982. United Nations convention on the Law of The sea. In: United Nations, Devision for Ocean Affairs and the Law of the Sea. United Nations, Montego Bay, Jamaica Resolution 2749 (XXV).

Utne, I.B., Sørensen, A.J., Schjølberg, I., 2017. 2017. Risk management of autonomous marine systems and operations, proceedings of the ASME 2017 36th international Conference on ocean, offshore and arctic engineering. OMAE Trondheim, Norway, p. V03BT02A020.

Vagia, M., Transeth, A.A., Fjerdingen, S.A., 2016. A Literature Review on the Levels of Automation during the Years. What Are the Different Taxonomies that Have Been

Proposed? Applied Ergonomics 53, Part a. pp. 190–202.

Valdez Banda, O.A., Goerlandt, F., Montewka, J., Kujala, P., 2015. A risk analysis of winter navigation in Finnish sea areas. Accid. Anal. Prev. 79, 100–116.

van Dorp, J.R., Merrick, J.R.W., 2011. On a risk management analysis of oil spill risk using maritime transportation system simulation. Ann. Oper. Res. 187 (1), 249–277.

Vanem, E., Puisa, R., Skjong, R., 2009. Standardized Risk Models for Formal Safety Assessment of Maritime Transportation. (43420). pp. 51–61.

Wang, J., Fan, Y.-T., 2008. Risk Analysis Based on the Ship Collision Modeling and Forecasting System, 2008 IEEE International Conference on Systems, Man and Cybernetics (SMC 2008). IEEE, Piscataway, NJ, USA, pp. 1517–1521 12-15 Oct. 2008.

Weng, J., Meng, Q., Qu, X., 2012. Vessel collision frequency estimation in the Singapore strait. J. Navig. 65 (2), 207–221.

Wrobel, K., Krata, P., Montewka, J., Hinz, T., 2016. Towards the development of a risk model for unmanned vessels design and operations. TransNav, the International Journal on Marine Navigation and Safety of Sea Transportation 10 (2), 267–274.

Wróbel, K., Montewka, J., Kujala, P., 2017. Towards the assessment of potential impact of unmanned vessels on maritime transportation safety. Reliab. Eng. Syst. Saf. 155–169.

Wróbel, K., Montewka, J., Kujala, P., 2018. System-theoretic approach to safety of remotely-controlled merchant vessel. Ocean Eng. 152, 334–345.

Xiao, F., Ligteringen, H., Van Gulijk, C., Ale, B., 2013. Nautical traffic simulation with multi-agent system for safety. In: IEEE Conference on Intelligent Transportation Systems, Proceedings, ITSC, 1245–1252.

Yang, X., Liu, X., Xu, T., 2011. Research of ship grounding prediction based on fuzzy theory. In: proceedings - 2011 international Conference of Information Technology, Computer Engineering and Management sciences, ICM 2011, pp. 91–94.

Zaman, M.B., Kobayashi, E., Wakabayashi, N., Khanfir, S., Pitana, T., Maimun, A., 2014. Fuzzy FMEA model for risk evaluation of ship collisions in the Malacca Strait: based on AIS data. J. Simulat. 8 (1), 91–104.

Zhang, D., Yan, X.P., Yang, Z.L., Wall, A., Wang, J., 2013. Incorporation of formal safety assessment and Bayesian network in navigational risk estimation of the Yangtze River. Reliab. Eng. Syst. Saf. 118, 93–105.

Zhang, W., Montewka, J., Goerlandt, F., 2015. Semi-qualitative method for ship collision risk assessment, safety and reliability: methodology and Applications - proceedings of the European safety and reliability conference. In: ESREL, pp. 1563–1572.

This page is intentionally left blank

Article 2

Thieme, C. A., Mosleh, A., Utne, I. B. & Hegde, J. Submitted. *Incorporating software failure in risk analysis – Part 1: Software functional failure mode classification. Submitted for review to Reliability Engineering and System Safety*.

This page is intentionally left blank

Incorporating software failure in risk analysis – Part 1: Software functional failure mode classification

Christoph A. Thieme[1,2*], Ali Mosleh[3,2], Ingrid B. Utne[1,2], and Jeevith Hegde[1]

[1]Norwegian University of Science and Technology (NTNU) Centre for Autonomous Marine Operations and Systems (AMOS), NTNU, Otto Nielsens Veg 10, 7491 Trondheim, Norway; [2] Department of Marine Technology, NTNU, Otto Nielsens Veg 10, 7491 Trondheim, Norway; [3] B. John Garrick Institute for the Risk Sciences, University of California, Los Angeles, 404 Westwood Plaza, Los Angeles CA90095, USA

*Corresponding author E-mail: Christoph.Thieme@ntnu.no

Abstract

Most advanced technological systems designed and operating today contain software. Risk analysis of these systems is necessary to ensure safe operation during their lifetime, but it is highly challenging to analyze risk related to software and the propagating effects on the system. Risk analysis often takes a functional approach. However, the functional level of software is not clearly defined; hence, several taxonomies for failure modes exist, of which none adhere fully to the designated level of analysis and none cover all failure modes from the other taxonomies.

Using a functional perspective of software, this article distinguishes between failure mode, failure cause, and failure effect as building block of software modeling for risk applications. Accordingly, 29 failure modes are identified to form a taxonomy with the following categories: functional, interaction, timing-related, and value-related failure modes. A case study demonstrates how these failure modes can be applied to software functions with different levels of detail. The failure mode taxonomy assists in identifying software failure modes, which provide input to the probabilistic analysis of software intensive systems presented in an accompanying article [1].

Keywords: Software failure mode; hazard identification; taxonomy; risk assessment; risk analysis

Acronyms

AROV	Autonomous Remotely Operated Vehicle
CAS	Collision avoidance system
CCF	Common Cause Failure
CPU	Central Processing Unit
CSRM	Context-based Software Risk Management
DFM	Dynamic Flowgraph Methodology

DP	Dynamic Positioning
ESD	Event Sequence Diagram
FM	Failure Mode (in the case study)
FMEA	Failure Mode and Effects Analysis
FT	Fault Tree
FTA	Fault Tree Analysis
NASA	National Aeronautics and Space Administration
OECD	Organization for Economic Co-operation and Development
STPA	System-Theoretic Process Analysis

1 Introduction

Risk assessment provides decision support in relation to risk in technological systems. One important step of risk assessment is the identification of hazardous events. Software is found today in almost any technological system. However, assessment of software intensive systems proves difficult with the traditional methods of risk assessment [2].

Examples of such systems are autonomous vehicles and vessels, which will be an essential part of future transportation systems [3]. Autonomous cars are being tested on the roads. In the maritime industry, autonomous ships are expected to operate within the next five years [4, 5]. It may be questioned whether the risks associated with these systems are too high for the authorities to approve and the public to accept the widespread use of such systems.

In the maritime industry, concerns are expressed with respect to the demonstration of safety of autonomous ships and the assessment of their control systems [6]. Hence, it is necessary to assess the risks associated with these systems. A risk assessment attempts to answer three questions to suggest measures to improve the system: *(i) what can go wrong, (ii) how likely is it that it will happen, and (iii) if it does happen, what are the consequences?* [7]. A risk analysis is the process to answer these questions.

Unlike hardware failures, software failures might lead to unanticipated effects that are not easily identified [8]. In addition, external interfaces and related failures should be considered to cover the whole spectrum of possible failures [8, 9].

To respond to the first question, the identification of potentially hazardous events or possible failure modes is an important part and is the first step of risk analysis of technological systems. A failure mode is the manner in which an item fails [10]. Failure modes need to be related to the context of operation [11]. This is particularly true for software failure modes. A generated numerical output that is legitimate and correct in general may be wrong and inappropriate for a specific situation (context).

A number of approaches have been introduced to cover the contribution of software to risk of systems. Examples are software failure mode effects analysis (FMEA) [e.g., 12, 13-18], dynamic flow graph methodology (DFM) [19-22], or simulations with failure mode injection [e.g., 23, 24, 25]. So far, risk assessments of autonomous marine systems [e.g., 26, 27-33] do not consider software failures in detail. Experts agree that software that is used under normal operational conditions can be analyzed from a functional point of view [34] with respect to risk.

Different taxonomies for failure modes of software exist [e.g., 11, 35]. However, the taxonomies do not adhere to one level of analysis consistently (e.g., the software system level, functional level, or code level). The result makes analyses and decision making for mitigating measures of software risks difficult.

The objective of this article is to identify and structure generic software functional failure modes into one consistent and comprehensive taxonomy. For this purpose, the concept of a software function is clarified. The generic failure modes can be combined with specific functions of a software system. The purpose is to provide a comprehensive and systematic basis for risk analysis of software-intensive systems, such as autonomous vehicles and ships. The generic set of failure modes may improve the failure mode identification process with respect to coherence and reduced time needed for such assessments.

An accompanying article, Thieme et al. [1] proposes a method for incorporating and assessing the effect of the software failure modes in risk analysis. The method relies on software functional representation and the generic software functional failure modes, defined in the present article.

The next section offers the necessary background on the methods for risk analysis of software systems and relevant software failure mode taxonomies. Section 3 defines and describes the concepts of software functions and functional decomposition of software. The failure mode taxonomy with a case study is proposed in Section 4. The last section discusses and concludes the work.

2 Existing Software Risk Analysis Methods

2.1 Risk analysis of software

In contrast to hardware systems, software fails mainly due to design and coding error. Software does not have a time-dependent failure rate, and different failure mechanisms and common cause failure (CCF) mechanisms cause failures of in software than those that cause hardware failures [34]. Generic failure modes can be used to describe most software-related failures, such that they can be implemented in risk analysis [35]. However, application-specific failure modes might be necessary in some cases [11, 35].

Ensuring software reliability is a key focus of design and development of most modern systems. However, a reliable software may still lead to hazardous situations and contribute to unacceptable levels of risk [36]. Several incidents show that correctly and reliably working software under some circumstances may lead to accidents (e.g., on space missions [37, 38], or in marine control systems [39]). Hence, reliability assessment methods are, to a limited extent, applicable for risk analysis. Reviews of the estimation of software reliability are summarized by Chu et al. [40], Yamada [41], and IEEE 1633 [42], for example.

Several methods and processes have been developed or adapted for the risk analysis of software. In addition, FMEA is a bottom-up analysis, which considers the failure of individual components and their associated effect on the overall system [10]. No formal process for software FMEA is defined [43], but the standard for hardware FMEA, IEC 60812 [10], is commonly used as basis. Several taxonomies exist for software failure modes for FMEA. These are described in more detail in Section 2.2.

Garrett et al. [19] and Guarro et al. [44] developed the DFM to assess the dependability and safety of software systems. The DFM is a two-step process: (i) build the model for the software system and (ii) analyze the model to build fault trees (FT). A DFM model is a directed graph with functional relations (the causality network) and conditions that trigger functional relations (conditional network). The software system is seen as a flow of information that is manipulated by different software functions. A timed FT is built in the second step of the DFM by assessing which conditions in the DFM model lead to the undesired top event of the FT.

Al Dabbagh [45] and Al Dabbagh and Lu [46] applied the DFM methodology to a networked control system of a communication network. Special sub-models have been developed to model reoccurring functions in such a network with a focus on the timing of functions of such a system (e.g., pre-processing times, waiting times, etc.). The DFM analyzes failures in relation to the flow of information and its timing. For example, missing operations or unanticipated function calls are not considered. The analysis uses little information from the software documentation. Failures that are related to datatype failures or other interactions between functions might be overlooked.

The National Aeronautics and Space Administration (NASA) [47, 48] has used DFM in their context-based software risk assessment methodology (CSRM). In the CSRM, critical mission stages are identified that include a risk-relevant software contribution. These mission stages are assessed with fault and event trees. Ref [48] has suggested using simple logic models, such as the fault tree analysis (FTA) for simple software systems or a high levels of modeling abstraction, while recommending DFM for more complex software systems that are also time-dependent behavior.

Aldemir et al. [49, 50] use Markov cell mapping combined with DFM. The method is capable of capturing the system behavior dynamically and discovering event sequences that otherwise are hard to identify by analyst. Aldemir et al. [49, 50] acknowledge that design errors might not be revealed with these methods. The Markov methods are state based, and analyzing different combinations of states requires setting up new models for each assessment. Since the Markov cell mapping is combined with DFM, the limitations of DFM apply to this combined method as well.

Li et al. [24] and Li [11] decompose software into functional units, and the failure of these functions are inserted in FT and event sequence diagrams (ESD) in the risk analysis. Only selected failure modes are implemented directly into the risk analysis, combining different levels of software analysis and decomposition. One notes that the concept of functional failure modes is not applied consistently, and certain failure modes are included that are not relevant for the functional level.

Wei [51] and Wei et al. [25] present a framework to include the risk contribution of software in risk analysis. The framework comprises four steps; input failure analyzer, operational profile builder, software propagation analyzer, and probabilistic risk assessment updater. The results of the analysis can be included in ESDs and FTs. The method is only applicable to existing software systems and not suitable for the design phase. In addition, the analysis does not make use of all the information that is typically available (e.g., the software specifications or safety requirements), which would reveal deficiencies with respect to these requirements.

Zhu [52] and Zhu et al. [23] build on the work of Wei [51] and include software failure in dynamic risk analyses, which also considers the timing of events relative to each other. Random software failures are *injected* in a dynamic model, and the simulation *reacts* to these failures. Associated faults and event trees are built automatically by the system. The software behavior and failures are represented in finite state machines. In their construct, the simulation model covers only selected failure modes and their influence on the dynamic behavior.

Leveson [53] and Leveson et al. [54] state that the systems-theoretic process analysis (STPA) is a hazard identification method that is also suitable for software. In their construct, hazards arise from insufficient control actions, that is, not providing a necessary control action, providing an unsafe control action, providing a potential control action too late, too early, or out of sequence, or providing a safe control action that is too short or too long [54]. Abdulkhaleq and Wagner [55] and Abdulkhaleq et al. [56] have extended the STPA for automated model checking of critical software applications, identifying potential hazardous situations from a software model and verifying that the taken control actions are safe.

Rokseth et al. [57] recommend combining FMEA and an adapted version of STPA to improve the identification, analysis, and verification of hazardous events and failure modes of dynamic positioning (DP) systems in ships and offshore oil and gas rigs. The STPA and FMEA were found to be complimentary, and a combination would be most suitable for complex and software intensive systems, such as DP. Positioning systems will be crucial for autonomous systems, such as ships [58, 59].

Gran [60] develop an influence network to assess the quality of the software process, the resulting quality of the software, and the associated risk. However, this approach does not consider the specific purpose of the software and the influence on the risk level or hazards that arise from the software. Therefore, it is not possible to identify and incorporate hazards in risk analysis.

Hewett and Seker [61] analyze the risk of embedded software systems with timed decisions tables. The approach is similar to DFM. Decision tables represent the software behavior, which is decomposed into functional modules. From the decision tables, timed FTs are built based on a predefined initiating event through backward reasoning. Similar to DFM, only failures that are related to a wrong value and the associated decisions are considered. Hence, it is not possible to identify failures and their contribution to the risk level that relate to unanticipated interaction of functions or datatype failures, for example.

Sadiq et al. [62] propose a software risk analysis using software FTA. The framework is intended for prioritizing testing and improvement of the software. It also highlights the necessity to consider the software requirements, modeling uncertainty, and possible errors in the analysis. However, the method does not allow for identifying events that might arise from within the software, such as interactions with other system components. The method only addresses the software system level.

2.2 Software Failure Mode Taxonomies

A literature search was conducted to identify existing failure mode taxonomies for software. A search on *IEEE Xplore*[1] and Scopus[2] was conducted. The keywords *software failure mode identification*, *software FMEA*, *software FMECA*, *software failure mode effect analysis*, and *software failure mode effect criticality analysis* were used in the search.

The search only covers publications since 2000, based on the assumption that these publications also reflect previous taxonomies. Publications that include relevant taxonomies have been closely investigated and have been selected for further analysis. Taxonomies that

[1] http://ieeexplore.ieee.org/; accessed Feb. 02, 2018.

[2] https://www.scopus.com/; accessed Dec. 08, 2017.

are the same in several publications are only assessed once. Eight publications, published between 2002 and 2014, include relevant taxonomies.

Li et al. [24] and Li [11] developed a functional failure mode taxonomy for software functions. They [11, 24] defined seven types of failure modes: functional, attribute, function set, timing, input/ output, multiple interaction, and support failure modes. Functional, attribute, and function set failure modes are summarized as *functional failure modes* for brevity [24]. Timing, input, output, multiple interaction, and support failure modes are *external failure modes*.

Input and output failure modes comprise failure modes that do not originate from the software function itself [11]. An input failure will lead to an output failure. Input and output failure modes can be further divided into *value-related failure modes* and *timing-related failure modes* [11].

Multiple interaction failure modes refer to communication through a common language to exchange information [11]. Support failure modes comprise failure modes related to hardware resources and the physical operating environment. Support failure modes, as described by Li [11], are not covered. These failure modes are related to physical failures and do not apply to the software functions. These failure modes fall in the category of failure causes of software, (c.f., Section 2.4).

The Organization for Economic Cooperation and Development (OECD) [35] has presented a taxonomy for hardware and software failure modes. The taxonomy builds on research by Li et al. [11, 24, 63], Authen et al. [64], Authen and Holmberg [65, 66], and Holmberg et al. [67], among others. It addresses different levels of the control system: overall system level, division level, instrumentation and control unit level, and instrumentation and control categories.

Ristord and Esmenjaud [14], Huang [68], Stadler and Seidl [15], Park [18], and Prasanna et al. [17] present their own adaptations of software FMEA. Each of them presents their own set of software failure modes that are considered. The literature offers a basis for identification of possible failure modes. However, a clear description and distinction of the targeted software level of abstraction for the taxonomies is absent. Only the taxonomy by OECD [35] attempts such a distinction.

2.3 Failure Mode Propagation

Failure propagation determines how a failure mode in one function will affect the software system [51]. Two main categories of failure propagation exist: CCF and cascade failures [35]. The CCF mechanisms affect several sub-systems such that the whole system fails. They occur under a specific set of conditions [35]. Cascading failure propagation occurs if one faulty output is the input to another function [35].

Propagation means that the failure is not masked or discovered and resolved during the execution of the program. Masking means a situation in which the software behavior produces the right output despite a failure during the execution. Multi-layer traps might conceal a failure through several sub-functions of the software [51]. Wei [51] derived a set of propagation mechanisms for software failure modes, which should be considered in risk analysis.

2.4 Failure Causes

Each failure mode may be attributable to one or more failure causes. [69]. The causes for software failures can be found in its specification, design, or implementation [14]. Moreover, NASA document [47] states additional causes to be parameter and data-entry errors and defects introduced during the removal of other defects. Ozarin [8, 9, 16, 43] highlight the necessity to consider the interaction of software-hardware interfaces when analyzing software, especially with respect to causes, such as bad input data or analog/digital converter failure. Stadler and Seidl [15] mention infinite loop, multi-process/thread/deadlock, counter rollover, numerical overflow/underflow/saturation, and finite precision error among others as potential failure causes.

3 Functional View of Software

A functional view of a system facilitates the specification of the software system and is advantageous in FMEA and model-based risk analysis methods [70]. A functional view of a software system is advantageous in analyzing operating software [34]. The term software system refers to the software program with its algorithms and implementation on the hardware. The software system can be decomposed in its functions. The purpose of a functional decomposition is to enable identification of relevant failure modes associated with each software function.

Figure 1 illustrates that software can be analyzed and broken down into functions with different levels of detail. Decomposing the software further will eventually lead to the software code level in an abstracted form, represented by pseudo code. Such a low level of decomposition is not covered by the taxonomy in this article.

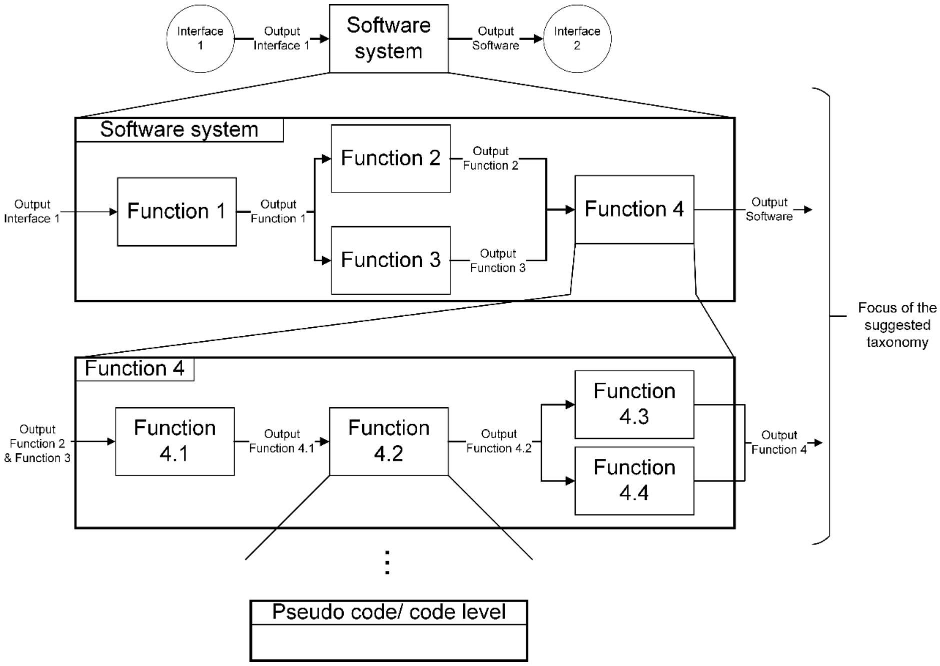


Figure 1 Different levels of software functions of a software system.

Beginning from the overall functional description, the software should be decomposed into sub-functions. These sub-functions describe what the software should do, not how it is implemented. In addition, EN14514 [70] gives guidance for the decomposition. Two factors determine the level of decomposition: design maturity [70] and the depth of the analysis of the software. Information for the decomposition can be extracted from the safety requirement specification (SRS), such as that defined by IEEE 830 [71]. If a functional decomposition has been executed during the software development phase, it should be used.

Figure 2 shows a generic function and its main elements once the desired level of breakdown and resolution is achieved. In addition, it shows where the different categories of failure modes can be applied. The process section is where the functional behavior and computation are executed, turning input into output. Function failure modes are associated with this part of the function.

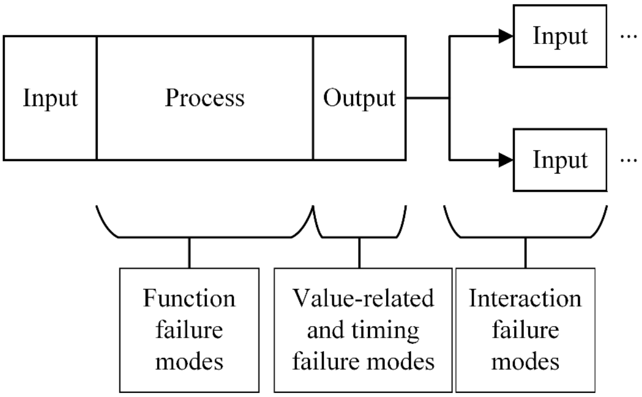

Input
Process
Output
Input
...
Input
...
Function failure modes
Value-related and timing failure modes
Interaction failure modes

Figure 2 Simplified view of a software function and its components underlying all levels. Developed and extended from [68].

The description of each function includes the function purpose, function process, necessary input and output produced, conditions of the function execution, requirements, constraints, and failure detection and correction mechanisms. Table 1 shows an exemplary datasheet for describing a function. The collection of all the information is necessary to determine the relevant failure modes. All information may not be available [34], but the as much as possible information should be used.

A function always has at least one output. This might be a numerical value, a binary value, or a function call to a specific function. Input is the output of another function or is given from external interfaces. An output of one function can be the input to several other functions. Each function might have several inputs and several outputs.

Input and output are associated with a datatype and an acceptable range. The range might be limited by the datatype, the acceptable value, or the set of meaning assigned to the values. The data format refers to the order of elements if the output is part of a data array or structure. The data rate describes a periodical output and its characteristics. Buffer refers to the type of buffer that is used for an output or input to collect data or events. Both the rate and buffer only need to be described if they are applicable. Both value-related and timing-related failure modes are associated with this part of a software function.

Functions in a software system are executed in a specified order. They might be executed periodically or on demand, depending on the result of the operation. Each function passes on information or calls another function. These interactions, represented by arrows, are associated with interaction failure modes. External interfaces are agents that interact with the software, such as other software systems, sensors, databases, or human operators through a human-machine interface.

Table 1 Exemplary datasheet for a software function.

ID:	**Datasheet for function**						**ID**
Function purpose	Short description of what the Function is to achieve						
Inputs	List and description of inputs received by the function						
	Input name	**Source**	**Data type**	**Data Format**	**Range**	**Rate**	**Buffer**
Outputs	List and description of outputs that the function produces						
	Input name	**Target**	**Data type**	**Data Format**	**Range**	**Rate**	**Buffer**
Conditions	Trigger conditions						
	Conditions to trigger other functions						
Process	Describe the behavior through formulas of input → output						
	Consider dependencies and sequence of operations						
Requirements	**Functional:**	Requirements related to the function itself (e.g., accuracy)					
	Non-functional	These can be requirements in relation to speed, security, safety, use of resources, etc.					
Constraints	Factors that limit the way a function could be implemented. Examples are regulatory constraints, hardware constraints, high order language requirements, signal handshake protocols, and criticality of the application.						
Failure detection and correction features	Measures that are implemented to detect, handle, and warn about software failures, such as control function, validity checks on the input, error handling system, etc.						

In a companion article [1] we propose a process to incorporate software in risk analysis. For this purpose, software functional failure modes are identified, and their effects at the software system level are analyzed. The results can be used as input in the risk analysis of a complex technological system. The decomposition of the software into functions is an essential part of the method proposed in [1].

4 Proposed Taxonomy

4.1 Procedure

To determine whether existing failure modes found in the literature are relevant for the functional level of software and the functional failure mode taxonomy for software, three questions were asked:

1. Does the presented failure mode fall into the definition of a failure mode?
2. If yes, does the failure mode fall into one of the failure mode categories, namely, interaction, function, value related, or timing related?
3. If yes, is the failure mode different from the failure modes identified previously?

If all questions were answered “yes”, the failure mode is included in the failure mode taxonomy. If a failure mode does not fulfill the definition of a failure mode or does not fit into one of the categories mentioned this failure mode is rejected. This is necessary to define an unambiguous and consistent failure mode taxonomy. Where it seems necessary, distinctions of similar failure modes are included to give more guidance for their use. These distinctions are labeled as refined failure modes.

Table 2 summarizes the contribution of the relevant publications identified in the literature screening. They contain relevant types of failure modes with respect to the scope of this article, which is the software functional level. All reviewed taxonomies cover value-related failure modes. All publications, except Prasanna et al. [17], cover timing-related failure modes. Except Wei [51], all publications consider interaction failure modes. Function failure modes are covered by only five publications.

Table 2 Summary of publications that form the basis for identification of the functional failure mode taxonomy of software.

Publication	Number of failure modes		Failures modes cover			
	Presented	Relevant	Function	Interaction	Timing-related	Value-related
Ristord and Esmenjaud, [14]	12	5	Yes	Yes	Yes	Yes
Li [11]	31	19	Yes	Yes	Yes	Yes
Wei [51]	12	12	No	No	Yes	Yes
Huang [68]	25	22	No	Yes	Yes	Yes
Stadler and Seidl [15]	21	17	Yes	Yes	Yes	Yes
OECD [35]	37	22	Yes	Yes	Yes	Yes
Park, 2014 [18]	21	14	Yes	Yes	Yes	Yes
Prasanna et al. [17]	11	10	No	Yes	No	Yes

Each of the presented publications has a different focus and therefore presents a different number of failure modes in each category, with different levels of detail. Most failure modes are presented in the OECD [35] study.

Several failure modes have been rejected for the proposed taxonomy. Stadler and Seidl [15] included *memory address errors* in their taxonomy. However, these are not relevant from a functional view since they are actually causes of a functional failure. Similarly, the failure modes *central processing unit (CPU) failure* [24], *memory failure* [24], *deadlock* [24], and *stop of operating system* [14] do not represent function failures and are considered a failure cause. *Interrupt induced failures* [35] and similarly *raised execution or interrupt* [15] already imply that they are a failure cause and not the failure mode. Hence, they were excluded. The failure mode *wrong task scheduling* [17] is a very coarse description and represents several interaction failure modes.

The failure modes *software aborts* [35], *hang/crash* [35], *program stop with/without clear message* [14], and *fail to return/complete* [15] were rejected since they represent effects of failure modes on the software system level. Table 3 summarizes the resulting failure mode taxonomy for function failure modes. Six failure modes were identified that address the functionality of a function.

Table 3 Taxonomy for function failure modes of software functions.

Failure mode	**Additional description**
Omission of a function/ missing operation	A function or a part of it is not executed.
Incorrect functionality	A function is not executing the intended actions.
Additional functionality	Extra non-specified operation in the function executed by the function.
No voting	Voting within the function is not carried out.
Incorrect voting	Voting within a function is not carried out according to specification.
Failure in failure handling	Detected failures are not handled appropriately.

4.2 Taxonomy

Table 4 summarizes the interaction failure modes between software functions. These failure modes reflect a failure of interaction between software functions. Seven failure modes were identified for the interaction between functions. Ten refined failure modes were identified for the interaction between software functions and external files or databases.

Table 4, Table 5, and Table 6 include the column *refined failure mode*. A refined failure mode represents a more detailed case of the failure mode. This was done to retain the knowledge presented in the literature, while classifying the failure modes generically.

Table 4 Taxonomy for interaction failure modes of software functions.

Failure mode	Refined failure mode	Additional description
Diverted/incorrect functional call		A wrong function is called after the current function is finished.
No call of next function		No further functions are called after the current function is finished.
No priority for concurrent functions		Functional calls for functions that need to be executed concurrently are given no priority
Incorrect priority for concurrent functions		Functions needed to be executed concurrently are given incorrect priority.
Communication protocol-dependent failure modes		Failure modes specific to a certain communication protocol that is used to interchange information between parts of the software system.
Unexpected interaction with input-output (IO) boards		Failure mode related to the interaction and spurious interaction with an IO board or an interface.
Failure of interaction with external files or databases	Wrong name	The name of the file/database is not correct.
	Invalid name/extension	The name entered for the file or database contains invalid symbols.
	File/ database does not exist	The file/ database name is specified correctly but the file/database does not exist.
	File/ database is open	The file/database is opened by another program and cannot be opened.
	Wrong/invalid file format	The file format is different from the expected file format.
	File head contains error	The file header information contains different information than required.
	File ending contains error	The file ending information contains different information than required.
	Wrong file length	The length of the file is different from the required/ expected length.
	File/database is empty	
	Wrong file/database contents	The information in the file/ database is different from the expected/ required information.

Table 5 summarizes the five failure modes related to timing-related failures. Five refined failure modes were identified for the timing of the output (i.e., too early or too late). Output rate failures form another category for four refined failure modes.

Table 5 Taxonomy for timing-related failure modes of software functions.

Failure mode	Refined failure mode	Additional description
Output provided	Too early	
	Too late	
	Spurious	Output provided when not requested or needed.
	Out of sequence	
	Not in time	No output is provided from the function.
Output rate failure	Too fast	
	Too slow	
	Inconsistent	
	Desynchronized	
Duration	Too long	Length of time the output is available.
	Too short	
Recurrent functions scheduled incorrectly		Periodically required output not delivered at the expected time.

Table 6 summarizes the 11 value-related failure modes of software functions. Four refined failure modes are found for the failure mode *incorrect value*. Five refined failure modes can be applied to data arrays or structures. Three refined failure modes are related to the validation of data.

Table 6 Taxonomy for value-related failure modes of software functions.

Failure mode	Refined failure mode	Additional description
No value		No value is provided.
Incorrect value	Too high	Value is higher than the expected and required value. This might be 1, the maximal allowable, or a higher increment of the value.
	Too low	Value is lower than the expected and required value.
	Opposite/inverse value	The value is the opposite or inverse of the expected value.
	Value is 0 (zero)	The value is zero instead of the expected value.
Value out of range	Datatype allowable range	
	Application allowable range	
Redundant/frozen value		The same value is produced constantly.
Noisy value/precision error		The values that are transferred are not precise enough.
Value with wrong datatype		
Non-numerical value	Not a number (NaN)	Values are transferred that are not interpretable by the software.
	Infinite	
	Negative infinite	
Elements in a data array/ structure	Too many	
	Too few	
	Data in wrong order	
	Data in reversed order	
	Enumerated value incorrect	Wrong element in the data array/ structure is addressed.
Correct value is validated as incorrect		
Incorrect value is validated as correct		
Data is not validated		Validity check is not executed.

4.3 Discussion

The challenge of identifying failure modes is that a clear distinction between failure mode, failure cause, and failure effect is difficult to achieve. The taxonomy proposed in this article attempts to clearly separate failure effects and causes from the functional failure modes.

Hence, failure modes, such as *incorrect realization of an attribute or function* [24] or *incorrect realization of a function* [35], were not included in the proposed taxonomy since they are considered to be failure causes, originating from the software programming (called realization) process.

Some failure modes have refined failure modes. For example, *Interaction with external files or databases* is refined by several sub-failure modes. Only identifying a *wrong interaction with the external files or databases,* is not a useful failure mode for further analysis, so it is necessary to specify how it is interacting wrongly.

Similarly, for timing-related failure modes, the *output rate failure* is rather vague. Hence, the refined failure modes *too slow, too fast, inconsistent*, and *desynchronized* were retained from the literature. Especially in the category *value-related failure modes*, several distinctions were made. The failure mode *incorrect value* would cover most of the failure modes. However, this is too generic in many cases. Therefore, refined failure modes for *incorrect value* were introduced. In addition, *non-numerical values* are differentiated since they will have a different effect on the software function than an incorrect numerical value. This adds more meaning to the failure modes and allows for application-specific failure mode assessment.

The adopted view on software is challenging in terms of the identification of a sufficiently low level of decomposition. The level of detail of software decomposition is dependent on the maturity of the software and the purpose of the analysis, such as a detailed risk study. A functional view allows the analysts to analyze the software in an early development stage since it is independent of the implementation. Especially, during early stage of development, the software documentation might be immature, and decomposition may only be possible at a higher level. Decomposition down to the code level is not recommended since even medium-sized software projects have several tens of thousands of lines of code.

5 Case Study

The presented taxonomy is generic in nature. The analysts need to give meaning to the failure modes for each function with the context of analysis. To demonstrate the application possibilities, a case study is included. Hegde et al. [72, 73] have presented an underwater collision avoidance system (CAS) based on safety envelopes and subsea traffic rules for an autonomous remotely operated vehicle (AROV).

The AROVs are tethered underwater robots that have a high level of autonomy in their operation. The underwater CAS provides decision support with respect to safe operation of the AROV. It is necessary to assure that the underwater CAS does not increase the level of risk. The underwater CAS receives data from a database and provides information for operational decision making. This section focuses on demonstrating the individual failure modes in an application setting. It represents parts of Steps 2 and 3 of the process in the accompanying article [1].

5.1 Functional decomposition

The functional hierarchy (Figure 3) identifies five functions for the software in the case study on the first level of decomposition. These functions are initialize underwater CAS, obtain data, determine suggested action, prepare renderer information, and display information. The underwater CAS is a rather small software with about 1,000 lines of code. Hence, it was decided that a decomposition to the first level is sufficient. As an example, *initialize underwater CAS* was decomposed to the second level. The functions on the second level are already close to pseudo code; thus, decomposing the function further would lead to code instructions.

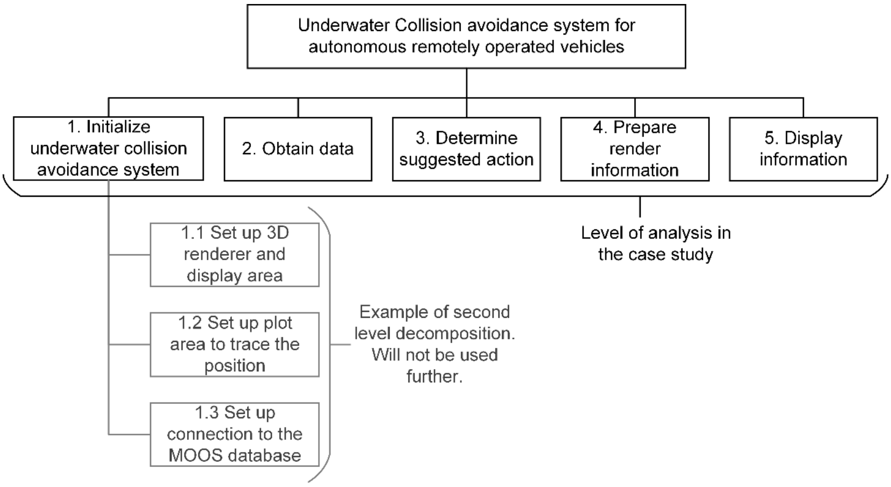

Underwater Collision avoidance system for autonomous remotely operated vehicles
1. Initialize underwater collision avoidance system
2. Obtain data
3. Determine suggested action
4. Prepare render information
5. Display information
Level of analysis in the case study
1.1 Set up 3D renderer and display area
1.2 Set up plot area to trace the position
1.3 Set up connection to the MOOS database
Example of second level decomposition. Will not be used further.

Figure 3 Functional decomposition of the underwater collision avoidance system.

Function 2, *obtain data,* serves as an example. Table 7 describes it in detail. This function is a suitable example since it covers a variety of output types and functional behaviors. The function polls the database with a frequency of 2 Hz for data on the AROV orientation, AROV position, AROV operational mode, and information on identified collision candidates. The database returns the requested values, and the function obtain data should make them available for the subsequent functions. The value for AROV orientation is received from the database in radians and will be converted to degrees in the function.

The case study demonstrates that the failures can be applied to an advanced software system. All four types of failure modes could be applied to the software function. The identification process shows how the different types of failure modes can be assessed with different levels of detail.

5.2 Application of the failure mode taxonomy

Table 8 presents the identified failure modes with the taxonomy for Function 2. The developer of the underwater CAS (the co-author) and a risk analyst (the first author) carried out the assessment. The table does not present all value-related failure modes. More value-related failure modes can be identified similarly to the ones identified in the table. A detailed list of all failure modes would add to the length of the table but not more insight on the identification of failure modes.

As stated previously software failure modes are context specific. Hence, the context is required to identify relevant failure modes for a function. The top of Table 8 defines the expected input and output for the example function *obtain data*. This sets the context for the failure mode identification.

The failure modes are applied based on the information found in the datasheet in Table 7. The information on the inputs and outputs is necessary for the assessment of value-related failure modes. Conditions describe the functional interactions and dependencies with other functions. Functional and non-functional requirements set the context for the assessment, such as acceptable timing delays or value inaccuracies.

The top part of Table 8 shows that it is not always possible to define expected values. They might be unknown due to the complexity of the function or the behaviour of the function over time. In other cases, the expected values are known due to the context. In the case of the function *obtain data*, the expected values are assumed to be known. The AROV is traveling in semi-autonomous mode, Mode 1, from the south to the north without any pitch or roll angle, corresponding to [0, 0, 0]. An object has been detected to the left of the AROV, corresponding to the envelope elements [66, 67, 76, 77]. The exact location of the case study is not relevant, only its accuracy.

Table 7 Datasheet for Function 2: *obtain data*. Abbreviations: :i – integer, :f – float, :s – string, N.A. - not applicable, MDb – MOOS database.

ID: F2	**Function 2: Obtain data**							**ID**
Function purpose	Send a request for updated information on the parameters: AROV position, AROV orientation, safety envelope collision information, AROV operation mode and make it available for subsequent functions.							
Inputs	**Input name**	**Source**	**Data type**	**Data Format**	**Range**	**Rate**	**Buffer**	**ID**
	AROV orientation from database	MDb	float	:f, :f, :f	0 to 2*pi	2 Hz	N.A.	MDb.O1
	AROV operational mode from database	MDb	int	:i	0 to 2	2 Hz	N.A.	MDb.O2
	AROV position from database	MDb	float	:f, :f, :f		2 Hz	N.A.	MDb.O3
	Information on identified collision candidates	MDb	String	:s, max. 64 elements	00-07, 10-17, 20-27, 30-37, 40-47, 50-57, 60-67, 70-77	2 Hz	N.A.	MDb.O4
Outputs	**Output name**	**Target**	**Data type**	**Data Format**	**Range**	**Rate**	**Buffer**	**ID**
	Request for AROV orientation	MDb	String			2 Hz	N.A.	F2.O1
	Request for AROV operational mode	MDb	String			2 Hz	N.A.	F2.O2
	Request for AROV position	MDb	String			2 Hz	N.A.	F2.O3
	Request for information on identified collision candidates	MDb	String			2 Hz	N.A.	F2.O4
	AROV orientation	F4	float	:f, :f ,:f		N.A.	N.A.	F2.O5
	AROV operational mode	F4	int	:i	0 to 2	N.A.	N.A.	F2.O6
	AROV position	F4	float	:f, :f, :f		N.A.	N.A.	F2.O7
	Information on identified collision candidates	F3	String	:s, max. 64 elements	00-07, 10-17, 20-27, 30-37, 40-47, 50-57, 60-67, 70-77	N.A.	N.A.	F2.O8
Conditions	Initiated by F1							F2.C1
	Initiated by F4.2 after first iteration							F2.C2
	Initiate F3							F2.C3
Function behavior	Send request for: AROV position, AROV orientation, AROV operational mode, and information on identified collision candidates.							F2.B1
	Convert AROV orientation from radians to degrees.							

	Store values of AROV position, AROV orientation, AROV operational mode, and information on identified collision candidates in the corresponding variables.		F2.B2 F2.B3
Requirements	**Functional:**	None	
	Non-functional	Poll MDb with 2 Hz	F2.NF1
Constraints	Successful connection to MDb in F1.		F2.Ct1
Failure detection and correction features	Request for non-existing data to the database returns an error message and does not return a value		F2.D1

Table 8 Failure mode identification for the function *obtain data* of the underwater collision avoidance system. Failure modes are highlighted in italics.

Expected input		
ID	**Name**	**Expected value**
MDb.O1	AROV orientation from database	[0,0,0]
MDb.O2	AROV operational mode from database	1
MDb.O3	AROV position from database	Correct (not further specified)
MDb.O4	Information on identified collision candidates	[66, 67, 76, 77]
Expected output		
ID	**Name**	**Expected value**
F2.O1	Request for AROV orientation	Correct request
F2.O2	Request for AROV operational mode	Correct request
F2.O3	Request for AROV position	Correct request
F2.O4	Request for information on identified collision candidates	Correct request
F2.O5	AROV orientation	[0,0,0]
F2.O6	AROV operational mode	1
F2.O7	AROV position	Correct (not further specified)
F2.O8	Information on identified collision candidates	[66, 67, 76, 77]
F2.C3	Initiate F3	-
ID	**Associated element**	**Failure mode**
Function failure modes		
FM1	F2	*Omission* of “Obtain data”, which is not executed.
FM2	F2.B1	*Omission* of requesting data, which means that data is not requested.
FM3	F2.B2	*Omission* of converting MDb.O1 to AROV orientation data, which means that the orientation is note executed.
FM4	F2.B3	*Incorrect functionality* of storing values in the corresponding variables, making them unavailable
FM5	F2.B2	*Additional functionality* while converting AROV orientation (e.g., conversion of AROV position)
FM6	F2.D1	*Failure in failure handling*, not detected that no value has been received
Interaction failure modes		
FM7	F2.C3	*Incorrect function call*, calling function 4 “Prepare render information,” skipping function 3 “Determine suggested action”
FM8	F2.C3	*No function call* to F3
FM9	F2.C3	*Incorrect priority for functions*, call function F4 “Prepare render information,” followed by function 3 “Determine suggested action”

Expected input		
ID	**Name**	**Expected value**
FM10	F2.B1	Unable to request information from the database (*communication protocol-dependent failure*)
FM11	F2.B1	*Request with wrong variable name* to the database for AROV position
Timing-related failure modes		
FM12	F2.O1	*Output provided too early*: Request for AROV orientation
FM13	F2.O1	*Output provided too late:* Request for AROV orientation
FM14	F2.O1	*Output provided too late (500 ms):* Request for AROV orientation
FM15	F2.O7	*Output provided spuriously:* AROV operational mode
FM16	F2.O8	*Output provided out of sequence:* F2.O8 provided before F2.O7
FM17	F2.O8	*Output not provided in time*: Information on identified collision candidates
FM18	F2.O1-F2.O4	*Output rate too fast: Requests to database* sent too fast
FM19	F2.O1-F2.O4	*Output rate too slow: Requests to database* sent too slow
FM20	F2.O1-F2.O4	*Inconsistent* rate for requests
Value-related failure modes		
FM21	F2.O7	*No value* for AROV position
FM22	F2.O7	*Incorrect value* for AROV position (not further defined)
FM23	F2.O6	*Incorrect value*, too high for AROV operational mode = 2
FM24	F2.O6	*Incorrect value*, too low for AROV operational mode = 0
FM25	F2.O5	*Incorrect* value, too high, AROV orientation [0, 0, -15]
FM26	F2.O5	*Incorrect* value, too high, AROV orientation [0, 0, -30]
FM27	F2.O7	*Incorrect* value, *Zero* for AROV position [0,0,0]
FM28	F2.O8	*Value out of application allowable range* for Information on identified collision candidates includes the value 68
FM29	F2.O6	*Value out of datatype range* for AROV operational mode = 2,147,483,648
FM30	F2.O8	*Frozen value* for Information on identified collision candidates (no collisions candidates detected)
FM31	F2.O7	*Imprecise value* for AROV position varying more than 1 m
FM32	F2.O6	*Wrong datatype* for AROV operational mode, string instead of int
FM33	F2.O8	*Too many elements*, 65, in Information on identified collision candidates
FM34	F2.O5	*Too few elements*, (two elements instead of three), in AROV orientation
FM35	F2.O7	*Data in wrong orde*r in AROV position [z, x, y] instead of [x, y, z]
FM36	F2.O5-F2.O8	*Incorrect value (no value) is validated as correct* and is output

The first column in Table 8 is labeled ID for identifier. Each failure mode that is identified needs to have an identifier to be able to trace the failure modes. The second column summarizes the element that is affected by the failure mode, that is, the variable, execution timing, part of the function block, or a functional transfer. In the third column, the failure mode is described and specified.

The applied failure modes from the presented taxonomy are marked explicitly in italics in Table 9. The case study demonstrates that different levels of detail can be applied to the identified failure modes, such as FM22, FM23, FM25, and FM26.

The ID FM22 describes just that the value is generally incorrect. With the background information and level of detail available, it is sufficient to describe it as incorrect. For FM23, a definite value can be associated since the expected value is known. Both FM25 and FM26 are a special case of a too-high value. Sometimes, it might be necessary to differentiate in incremental steps since different values imply different interpretations of the failure mode and may lead to different risk contributions. Similarly, for timing, different levels of detail can be applied (i.e., FM13 and FM14). With a too-late value of 500 ms, FM14 is a refined version of FM13.

5.3 Discussion

The case study demonstrates how failure modes can be identified for different elements of a function. It demonstrates how several of the failure modes can be applied to the functional level of a software system. Not all failure modes could be applied and demonstrated, because not all failure modes were relevant for the case study, or there would have been a level of repetition of similar failure modes. However, application of the other failure modes is similar to the example laid out. The risk analysts along with software developers should be able to apply the failure modes in a manner that is relevant for the context. This is only possible if the analysts have a common understanding of the software system and the associated terminology.

How different levels of detail can be integrated into the identification and application of failure modes is demonstrated. For example, value-related can be described very generically as incorrect, or with a specific value, or within a specific range of values. This implies that the taxonomy is applicable during different project phases, such as the preliminary design, or detailed design.

One shortcoming of the case study is that the underwater CAS has not been developed according to a software development standard. Hence, the amount of information documented was limited. However, the main developer of the program co-authored this article and provided additional information when necessary.

6 Discussion and Conclusion

This article presents a functional failure mode taxonomy for software functions of a software system. No clear definition of the functional software level and associated failure modes exists so far. This article defines and clarifies the concept of software functions and associated failure modes for software systems. The taxonomy was synthesized from the literature and suited to the functional view taken in this article. The application of the failure mode taxonomy was tested on an actual software program. Application of the taxonomy contributes to an improved identification of software function failure modes and contributes to a systematic and thorough software failure mode identification process.

A functional view makes the analysis scalable, modular, and is appropriate for reliability and risk analysis. The system can be broken down to the desired level of detail and based on the availability of information at a given phase in the software life cycle. A functional analysis can be carried out in an early stage of development; hence, the failure modes can be identified and used from an early stage of development. The immediate effect on the software output might not be derived directly from the functional failure modes; therefore, failure effect propagation is needed.

The application of the proposed functional failure mode taxonomy for identification of failure modes is as time-consuming as it is for similar processes. However, having a generic taxonomy allows identifying failure modes more efficiently. The obtained set of failure modes for software functions will be more comprehensive, and the result may justify the effort. A computer-aided tool could be used for the assessment to reduce the work associated with the documentation and to focus on the identification process.

Applying the failure mode taxonomy from early stages of development identifies areas that need special attention during requirement specifications, testing, and verification activities. Since the focus is on the software functions, it can be analyzed before the programming of the software starts.

The accompanying article [1] presents a process for incorporating software in risk analysis. This process uses the failure mode taxonomy and analysis of the effect of the software failure modes on the external interfaces. These identified effects may be included in risk analysis.

The proposed taxonomy only considers the functional level of software. In the future, it might be useful to identify failure modes on levels such as the software system level or the code level and clearly define these, building on and extending the work described by OECD [35].

Acknowledgments

C. A. Thieme, and I. B. Utne appreciate the support of the Research Council of Norway through the Centres of Excellence funding scheme, Project number 223254 – NTNU AMOS.

J. Hegde has been supported by the Research Council of Norway, Statoil and TechnipFMC through the research project Next Generation Subsea Inspection, Maintenance and Repair Operations, 234108/E30 and associated with NTNU AMOS 223254.

C. A. Thieme acknowledges the financial support from Norges tekniske høgskoles fond, who supported this research with a scholarship for a research exchange with the B. John Garrick Institute for the Risk Sciences at the University of California in Los Angeles.

References

[1] Thieme CA, Mosleh A, Utne IB, Hegde J. Incorporating Software Failure in Risk Analysis – Part 2: Risk Modeling Process and Case Study. Submitted for review to Reliabilty Engineering and System Safety. Submitted: pp.
[2] Mosleh A. Pra: A Perspective on Strengths, Current Limitations, and Possible Improvements. Nuclear Engineering and Technology. 2014;46: pp. 1-10.
[3] Marr B. The Future of the Transport Industry - Iot, Big Data, Ai and Autonomous Vehicles. 2017; https://www.forbes.com/sites/bernardmarr/2017/11/06/the-future-of-the-transport-industry-iot-big-data-ai-and-autonomous-vehicles/#2b854d791137; Accessed: 21.02.2018
[4] Kongsberg Maritime. Yara and Kongsberg Enter into Partnership to Build World's First Autonomous and Zero Emissions Ship. 2017; https://www.km.kongsberg.com/ks/web/nokbg0238.nsf/AllWeb/98A8C576AEFC85AFC125811A0037F6C4?OpenDocument; Accessed: 27.07.2017
[5] Kongsberg Maritime. Automated Ships Ltd and Kongsberg to Build First Unmanned and Fully Autonomous Ship for Offshore Operations. 2016; https://www.km.kongsberg.com/ks/web/nokbg0238.nsf/AllWeb/65865972888D25FAC125805E00281D50?OpenDocument; Accessed on: 24.04.2018
[6] Nautilus Federation. Report of a Survey on What Maritime Professionals Think About Autonomous Shipping. Regulatory scoping exercise for the use of maritime autonomous surface ships (MASS). London, UK: International Maritime Organization, Maritime Safety Committee; 2018.
[7] Kaplan S, Garrick BJ. On the Quantitative Definition of Risk. Risk Analysis. 1981;1: pp. 11-27.
[8] Ozarin NW. The Role of Software Failure Modes and Effects Analysis for Interfaces in Safety- and Mission-Critical Systems. 2008 IEEE International Systems Conference Proceedings, SysCon 2008, April 7, 2008 - April 10, 2008. Montreal, QC, Canada: Inst. of Elec. and Elec. Eng. Computer Society; 2008. pp. p 200-207.
[9] Ozarin NW. Applying Software Failure Modes and Effects Analysis to Interfaces. Annual Reliability and Maintainability Symposium 2009. pp. 533-538.
[10] IEC EN. En Iec 60812: Analysis Techniques for System Reliability – Procedure for Failure Mode and Effects Analysis (Fmea). Brussels, Belgium: International Electrotechnical Commission, European Committee for Electrotechnical Standardization; 2006.
[11] Li B. Integrating Software into Pra (Probabilistic Risk Assessment) [Monograph]. College Park, Md: University of Maryland; 2004.
[12] Reifer DJ. Software Failure Modes and Effects Analysis. IEEE Transactions on Reliability. 1979;R-28: pp. 247-249.
[13] Goddard PL. Software Fmea Techniques. Proceedings of the Annual Reliability and Maintainability Symposium. 2000: pp. 118-123.
[14] Ristord L, Esmenjaud C. Fmea Performed on the Spinline3 Operational System Software as Part of the Tihange 1 Nis Refurbishment Safety Case. Cnra/Csni Workshop On Licensing And Operating Experience Of Computer-Based I&C Systems. Hluboka nad Vltavou, Czech Republic: NEA/CSN/OECD; 2002. pp. 37 - 50.

[15] Stadler JJ, Seidl NJ. Software Failure Modes and Effects Analysis. 59th Annual Reliability and Maintainability Symposium, RAMS 2013, January 28, 2013 - January 31, 2013. Orlando, FL, United states: Institute of Electrical and Electronics Engineers Inc.; 2013.

[16] Ozarin NW. Bridging Software and Hardware Fmea in Complex Systems. 2013 Proceedings Annual Reliability and Maintainability Symposium (RAMS) 2013. pp. 1-6.

[17] Prasanna KN, Gokhale SA, Agarwal R, Chetwani RR, Ravindra M, Bharadwaj KM. Application of Software Failure Mode and Effect Analysis for on-Board Software. 2014 International Conference on Advances in Computing, Communications and Informatics (ICACCI) 2014. pp. 684-688.

[18] Park GY, Kim DH, Lee DY. Software Fmea Analysis for Safety-Related Application Software. Annals of Nuclear Energy. 2014;70: pp. 96-102.

[19] Garrett CJ, Guarro SB, Apostolakis GE. The Dynamic Flowgraph Methodology for Assessing the Dependability of Embedded Software Systems. IEEE Transactions on Systems, Man, and Cybernetics. 1995;25: pp. 824-840.

[20] Yau MK, Dixon S, Guarro SB. Applications of the Dynamic Flowgraph Methodology to Dynamic Modeling and Analysis. 11th International Probabilistic Safety Assessment and Management Conference and the Annual European Safety and Reliability Conference 2012, PSAM11 ESREL 2012, June 25, 2012 - June 29, 2012. Helsinki, Finland: Probablistic Safety Assessment and Management (IAPSAM); 2012. pp. 606-615.

[21] Karanta I. Implementing Dynamic Flowgraph Methodology Models with Logic Programs. Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability. 2013;227: pp. 302-314.

[22] Guarro SB, Yau MK, Ozguner U, Aldemir T, Kurt A, Hejase M, et al. Formal Framework and Models for Validation and Verification of Software-Intensive Aerospace Systems. AIAA Information Systems-Infotech At Aerospace Conference, 2017, January 9, 2017 - January 13, 2017. Grapevine, TX, USA: American Institute of Aeronautics and Astronautics Inc, AIAA; 2017.

[23] Zhu D, Mosleh A, Smidts C. A Framework to Integrate Software Behavior into Dynamic Probabilistic Risk Assessment. Reliability Engineering & System Safety. 2007;92: pp. 1733-1755.

[24] Li B, Li M, Ghose S, Smidts C. Integrating Software into Pra. Issre 2003: 14th International Symposium on Software Reliability Engineering, Proceedings. 2003: pp. 457-467.

[25] Wei YY, Rodriguez M, Smidts CS. Probabilistic Risk Assessment Framework for Software Propagation Analysis of Failures. Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability. 2010;Vol. 224: pp. 113-135.

[26] Brito MP, Griffiths G. A Bayesian Approach for Predicting Risk of Autonomous Underwater Vehicle Loss During Their Missions. Reliability Engineering & System Safety. 2016;146: pp. 55-67.

[27] Brito MP, Griffiths G. Autonomy: Risk Assessment. In: Dhanak MR, Xiros NI, editors. Springer Handbook of Ocean Engineering. Berlin Heidelberg: Springer International Publishing; 2016. pp. 527-544.

[28] Rødseth ØJ, Burmeister H-C. Risk Assessment for an Unmanned Merchant Ship. TransNav, the International Journal on Marine Navigation and Safety of Sea Transportation. 2015;9: pp. 357-364.

[29] Kretschmann L, Rødseth ØJ, Tjora Å, Sage Fuller B, Noble H, Horahan J. D9.2: Qualitative Assessment. Maritime Unmanned Navigation through Intelligence in Networks. 1.0 ed. 2015. pp. 45.

[30] Kretschmann L, Rødseth ØJ, Sage Fuller B, Noble H, Horahan J, McDowell H. D9.3: Quantitative Assessment. Maritime Unmanned Navigation through Intelligence in Networks. 2015.

[31] Jensen F. Hazard and Risk Assessment of Unmanned Dry Bulk Carriers on the High Seas. Hamburg: Technische Universität Hamburg Harburg; 2015.

[32] Thieme CA, Utne IB, Schjølberg I. A Risk Management Framework for Unmanned Underwater Vehicles Focusing on Human and Organizational Factors Proceedings of the ASME 2015 34th International Conference on Ocean, Offshore and Arctic Engineering OMAE2015. St. John’s, NL, Canada: ASME; 2015.

[33] Thieme CA, Utne IB. A Risk Model for Autonomous Marine Systems and Operation Focusing on Human–Autonomy Collaboration. Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability. 2017;231: pp. 446-464.

[34] Chu T-L, Martinez-Guridi G, Yue M, Samanta P, Vinod G, Lehner J. Workshop on Philosophical Basis for Incorporating Software Failures in a Probabilistic Risk Assessment. Digital System Software PRA. Brookhaven National Laboratory; 2009. pp. 188 pages.

[35] OECD. Failure Modes Taxonomy for Reliability Assessment of Digital I&C Systems for Pra. Paris: Nuclear Energy Agency; 2014. pp. 135.

[36] Garrett CJ, Apostolakis G. Context in the Risk Assessment of Digital Systems. Risk Analysis. 1999;19: pp. 23-32.

[37] Albee A, Battel S, Brace R, Burdick G, Burr P, Casani J, et al. Report on the Loss of the Mars Polar Lander and Deep Space 2 Missions. Pasadena, CA, USA: JPL Special Review Board; 2000.
[38] Tolker-Nielsen T. Exomars 2016 - Schiaparelli Anomaly Inquiry. Paris, France2017.
[39] Marine Accident Investigation Branch. Sbs Typhoon Contact in Aberdeen Harbour, 26 February 2011. 2011.
[40] Chu T-L, Yue M, Martinez-Guridi G, Lehner J. Review of Quantitative Software Reliability Methods. U.S. Nuclear Regulatory Commission, Office of Nuclear Regulatory Research, Division of Risk Analysis; 2010. pp. 98.
[41] Yamada S. Software Reliability Modeling - Fundamentals and Applications. Tokyo, Heidelberg, New York, Dordrecht, London: Springer Japan; 2014.
[42] IEEE. Ieee Std 1633-2016: Ieee Recommended Practice on Software Reliability. New York, NY, USA: IEEE Reliability Society; 2016. pp. 261.
[43] Ozarin NW. Developing Rules for Failure Modes and Effects Analysis of Computer Software. SAE Advances in Aviation Safety Conference - 2003 Aerospace Congress and Exhibition, September 8, 2003 - September 11, 2003. Montreal, QC, Canada: SAE International; 2003.
[44] Guarro SB, Yau MK, Motamed M. Development of Tools for Safety Analysis of Control Software in Advanced Reactors. Washington DC: ASCA Inc.; 1996. pp. 147.
[45] Al-Dabbagh AW. Dynamic Flowgraph Methodology for Reliability Modelling of Networked Control Systems [Master of Applied Science Thesis]. Oshawa, ON, Canada: University of Ontario Institute of Technology; 2009.
[46] Al-Dabbagh AW, Lu L. Reliability Modeling of Networked Control Systems Using Dynamic Flowgraph Methodology. Reliability Engineering & System Safety. 2010;95: pp. 1202-1209.
[47] Stamatelatos M, Dezfuli H, Apostolakis G, Everline CJ, Guarro SB, Mathias D, et al. Probabilistic Risk Assessment Procedures Guide for Nasa Managers and Practitioners. Washington D.C.: National Aeronautics and Space Administration; 2011. pp. 431.
[48] Guarro SB, Yau MK, Dixon S. Context-Based Software Risk Model (Csrm) Application Guide. Version 1.0 ed. Washington, D.C. 20546: ASCA Inc.; 2013.
[49] Aldemir T, Guarro SB, Kirschenbaum J, Mandellil D, Mangan LA, Bucci P, et al. A Benchmark Implementation of Two Dynamic Methodologies for the Reliability Modeling of Digital Instrumentation and Control Systems. Washington DC: Office of Nuclear Regulatory Research; 2009.
[50] Aldemir T, Guarro SB, Mandelli D, Kirschenbaum J, Mangan LA, Bucci P, et al. Probabilistic Risk Assessment Modeling of Digital Instrumentation and Control Systems Using Two Dynamic Methodologies. Reliability Engineering & System Safety. 2010;95: pp. 1011-1039.
[51] Wei YY. A Study of Software Input Failure Propagation Mechanisms. College Park, MD: University of Maryland; 2006.
[52] Zhu D. Integrating Software Behavior into Dynamic Probabilistic Risk Assessment. Collage Park, Md: University of Maryland; 2005.
[53] Leveson NG. A New Accident Model for Engineering Safer Systems. Safety Science. 2004;42: pp. 237-270.
[54] Leveson NG, Fleming CH, Spencer M, Thomas J, Wilkinson C. Safety Assessment of Complex, Software-Intensive Systems. SAE International Journal of Aerospace. 2012;5: pp. 233-244.
[55] Abdulkhaleq A, Wagner S. Integrated Safety Analysis Using Systems-Theoretic Process Analysis and Software Model Checking. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) 2015. pp. 121-134.
[56] Abdulkhaleq A, Wagner S, Leveson N. A Comprehensive Safety Engineering Approach for Software-Intensive Systems Based on Stpa. Procedia Engineering. 2015. pp. 2-11.
[57] Rokseth B, Utne IB, Vinnem JE. A Systems Approach to Risk Analysis of Maritime Operations. Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability. 2016: pp. 1748006X16682606.
[58] Advanced Autonomous Waterborne Applications. Remote and Autonomous Ships - the Next Steps. In: Laurinen M, editor. Adavanced Autonomous Waterborne Applications London, UK 2016. pp. 88.
[59] Bureau Veritas. Guidelines for Autonomous Shipping. Paris, France2017.
[60] Gran BA. The Use of Bayesian Belief Networks for Combining Disparate Sources of Information in the Safety Assessment of Software Based Systems. Trondheim, Norway: NTNU - Norwegian University of Science and Technology; 2002.
[61] Hewett R, Seker R. A Risk Assessment Model of Embedded Software Systems. 2005 29th Annual IEEE/NASA Software Engineering Workshop, SEW'05, April 6, 2005 - April 7, 2005. Greenbelt, MD, USA: Institute of Electrical and Electronics Engineers Computer Society; 2005. pp. 142-149.

[62] Sadiq M, Ahmad MW, Rahmani MKI, Jung S. Software Risk Assessment and Evaluation Process (Sraep) Using Model Based Approach. ICNIT 2010 - 2010 International Conference on Networking and Information Technology 2010. pp. 171-177.

[63] Li B, Li M, Smidts C. Integrating Software into Pra: A Test-Based Approach. Risk Analysis. 2005;25: pp. 1061-1077.

[64] Authen S, Björkman K, Holmberg J-E, Larsson J. Guidelines for Reliability Analysis of Digital Systems in Psa Context — Phase 1 Status Report. Roskilde, Denmark: Nordik Nuclear Safety Research (NKS); 2010. pp. 32.

[65] Authen S, Holmberg J-E. Reliability Analysis of Digital Systems in a Probabilistic Risk Analysis for Nuclear Power Plants. Nuclear Engineering and Technology. 2012;44: pp. 471-482.

[66] Authen S, Holmberg J-E. Guidelines for Reliability Analysis of Digital Systems in Psa Context — Phase 3 Status Report. Roskilde Denmark: Nordic nuclear safety research (NKS); 2013. pp. 61.

[67] Holmberg J-E, Authen S, Amri A. Development of Best Practice Guidelines on Failure Modes Taxonomy for Reliability Assessment of Digital Ic Systems for Psa. 11th International Probabilistic Safety Assessment and Management Conference and the Annual European Safety and Reliability Conference 2012, PSAM11 ESREL 2012, June 25, 2012 - June 29, 2012. Helsinki, Finland: Probablistic Safety Assessment and Management (IAPSAM); 2012. pp. 1887-1894.

[68] Huang B, Zhang H, Lu M. Software Fmea Approach Based on Failure Modes Database. 2009 8th International Conference on Reliability, Maintainability and Safety 2009. pp. 749-753.

[69] Ozarin NW, Siracusa M. A Process for Failure Modes and Effects Analysis of Computer Software. Proceedings of the Annual Reliability and Maintainability Symposium. 2003. pp. 365-370.

[70] EN. Ns-En14514: Space Engineering Standards - Functional Analysis. Brussels, Belgium: European Committee for Standardization; 2004.

[71] IEEE. Ieee 830: Recommended Practice for Software Requirements Specification. New York, NY, USA: IEEE; 2009.

[72] Hegde J, Henriksen EH, Utne IB, Schjølberg I. Development of Safety Envelopes and Subsea Traffic Rules for Autonomous Remotely Operated Vehicles. Submitted to: Journal of Loss Prevention in the Process Industries. submitted: pp. 133-155.

[73] Hegde J. Tools and Methods to Manage Risk in Autonomous Subsea Inspection, Maintenance and Repair Operations. Trondheim, Norway: Norwegian University of Science and Technology (NTNU); 2018.

This page is intentionally left blank

Article 3

Thieme, C. A., Mosleh, A., Utne, I. B. & Hegde, J. Submitted. *Incorporating software failure in risk analysis – Part 2: Risk analysis process and case study. Submitted for review to Reliability Engineering and System Safety*.

This page is intentionally left blank

Incorporating software failure in risk analysis — Part 2: Risk modeling process and case study

Christoph A. Thieme[1,2*], Ali Mosleh[3,2], Ingrid B. Utne[1,2], and Jeevith Hegde[1]

[1]Norwegian University of Science and Technology (NTNU) Centre for Autonomous Marine Operations and Systems (AMOS), NTNU, Otto Nielsens Veg 10, 7491 Trondheim, Norway; [2] Department of Marine Technology, NTNU, Otto Nielsens Veg 10, 7491 Trondheim, Norway; [3] B. John Garrick Institute for the Risk Sciences, University of California, Los Angeles, 404 Westwood Plaza, Los Angeles CA90095, USA
*Corresponding author E-mail: Christoph.Thieme@ntnu.no

Abstract

Most advanced technological systems contain a software component. With the advent of autonomous cars, drones, and ships, the complexity of these systems is increasing. One challenge lies in analyzing risk and its mitigation, as the incorporation of software failures currently proves difficult.

This paper is a follow-up article by Thieme et al. [1] and presents a method for the analysis of functional software failures, their propagation, and incorporation of the results in traditional risk analysis methods, such as fault trees, event trees, or event sequence diagrams. A case study focusing on a decision support system for an autonomous remotely operated vehicle working on a subsea oil and gas production system demonstrates the applicability of the proposed process. The results of the case study are used to derive software and system improvement measures.

Keywords: Software failure; risk analysis; propagating effects; autonomy

Acronyms

3D	Three Dimensional
AM	Active mode
AROV	Autonomous Remotely Operated Vehicle
CAS	Collision Avoidance System
DFM	Dynamic Flowgraph Methodology
ET	Event Tree
F	Function (in the case study)
FM	Failure Mode (in the case study)
FMEA	Failure Mode and Effects Analysis
FT	Fault Tree
FTA	Fault Tree Analysis
MOOS	Mission Oriented Operating Suite
MDb	Mission Oriented Operating Suite Database
PM	Passive Mode
SM	Sporadic mode
SRS	Software Requirements Specification
STPA	System-Theoretic Process Analysis
UI	User Input
US	User Screen

1 Introduction

Risk assessment provides decision support in relation to risk in technological systems. The aim is to identify and analyze hazardous events and critical failures and evaluate safeguards and tolerance to mitigate risk [2]. Risk analysis is the process to understand and determine the level of risk [3]. Today, software is found in almost all technological systems.

In the future, autonomous vehicles and vessels may be an essential part of the transportation system [4]. Autonomous systems will be confronted with various operational situations, involving several hazards that might not all be foreseeable for the system designer or analyst. Autonomous marine systems, such as ships [5-8] or autonomous underwater vehicles [9], are under development or exist already. These systems rely heavily on software.

Current methods applied in risk analysis, such as fault trees (FT) and event trees (ET), cannot reflect the interaction of complex software-based systems sufficiently [10]. Human, hardware, and software interfaces need to be considered to cover the whole spectrum of possible failures [11, 12].

Several challenges arise when attempting to analyze the risk contribution from software. Software might be reliable in the sense that it is executing the programmed actions correctly. However, the software might act reliably in a situation where the action might be considered unsafe [13]. Software behaves deterministically (i.e., software failures will always manifest under the same circumstances). Probabilistic methods can be used since there is uncertainty with respect to the knowledge that software is free from errors and that it will not exhibit any failures [14].

The objective of this article is to propose a process that may be used to identify hazardous events from software and analyze potential propagating effects on the overall system. The results from the process may be incorporated into risk analysis in a meaningful manner for further risk analysis. A functional view on software allows for flexible risk modeling, a solution-independent analysis of the events and effects, and a common foundation for communication between risk analysts and domain experts.

The proposed process can be used to identify necessary modifications and requirements for the software system, during the design, development, use, and modification stages in the software life cycle. In addition, it is possible to analyze how the software handles propagating failures caused by other components of the system, such as sensors and human operators. The case study in this article demonstrates the usability of the process.

This article builds on the background and results from the accompanying article [1], which provides a taxonomy for functional failure modes of software and the necessary foundations for the process proposed in this article. This article describes the suggested process for incorporating software failures in risk analysis. The process is qualitative; a quantification is not attempted yet.

A review of the relevant literature for software risk analysis and modeling approaches is presented in Section 2. This is followed by the developed and adapted process in Section 3. Section 4 exemplifies each step of the process. Sections 5 concludes this article.

2 Requirements for a Process Incorporating Software in Risk Analysis

A brief overview of current state-of-the-art methods to incorporate software into risk analysis is given in the accompanying article [1]. Software system is used to describe the whole software program with its algorithms and implementation on the hardware. This section presents a proposed set of requirements that were used as a guideline for developing the risk modeling process presenting in this article.

Garrett and Apostolakis [13] identified error forcing contexts, which will lead to software failure. They defined three abilities that a process should have: *represent all those states of the system that are deemed to be hazardous*, *model the functional and dynamic behavior of the software in terms of transitions between states of the system*, and *given a system failure event, identify the system states that preceded it*.

Hewett and Seker [15] identified four modeling properties of a risk analysis including software:

1. *Represents structures and (temporal) behaviors of the whole system (together with its interactions with external environments);*
2. *Supports the evolution of software*;
3. *Provides modularity and building-block capabilities to cope with scalability issues;*
4. *Offers systematic mechanisms to facilitate automated deduction and inference reasoning for risk assessment*.

Chu et al. [14] collected information from an expert panel on risk analysis of software systems. They agreed that a method incorporating software risk should account for different types of bugs and consider fault tolerant mechanisms and all available information on the software. Dependencies between hardware and software need to be included in the analysis.

A risk assessment shall answer three questions: *(i) What can go wrong? (ii) How likely is it that this will happen? (iii) If it does happen, what are the consequences?* [16]. Risk analysis is the *process to comprehend the nature of risk and to determine the level of risk* [3]. These definitions and the considerations above give input to the requirements for the process incorporating software in risk analyses in Table 1. If the questions are addressed by a process is assessed through R1 thru R3 and R7. The proposed process does not cover R7 and focuses on the quantitative process.

Table 1 Requirements for a process incorporating software in risk analysis, based on input from [13-15].

Requirement		Description
R1	Identify failure modes	The process shall enable the analyst to identify failure modes that might lead to unwanted consequences in the context.
R2	Identify possible failure causes	The risk model developed in the process shall assist in the identification of possible failure causes and sources in case of a failure.
R3	Identify consequences of failure modes	The process shall enable the analyst to trace software failure modes through risk scenarios leading to adverse consequences.
R4	Represent functional behavior	The risk model developed in the process shall reflect the functional behavior and constraints of the software including different states and transition between the states.
R5	Represent temporal behavior	The risk model developed in the process shall reflect time-related behavior, requirements, limitations, and states.
R6	Represent context of use	The risk model developed in the process shall include required contextual and overall constraints, hardware, software, and human interactions.
R7	Quantify likelihood of consequences	The process shall contain mechanisms for the quantification of failure modes and associated consequences.
R8	Be modular	The risk model developed in the process as well as the process shall be modular, such that changes in software modules can be reflected.
R9	Be scalable	The risk model developed in the process shall be scalable, such that different levels of detail can be addressed and that software systems of different sizes can be analyzed.
R10	Make use of all available information	The process shall use all available information to build and analyze the risk model developed in the process.
R11	Be applicable throughout the software life cycle	The process shall be appropriate throughout the lifecycle of the software and aide in decision making.

Requirements R6, R8, and R9 address features that a risk process to incorporate software in a risk analysis should exhibit in terms of the risk model developed in the process. Requirement R10 refers to the use of information for the process, while R11 shall assure that the process is applicable during the life of the software.

The requirements may be addressed using a functional perspective on the software, which makes it scalable and suitable for failure mode analysis [14, 17]. The discussion, Section 5, uses these requirements to highlight the features of the proposed process in comparison to existing methods and processes.

3 Process for Incorporating Software in Risk Assessment

Figure 1 summarizes the steps in the proposed process in this article and sets it in the context of the generic risk management framework presented by ISO 31000 [18]. Steps 2 to 4 are the main contribution from this article and the accompanying article [1], being improved and novel. The sections detail each of the steps, as indicated in the figure. Communication between different stakeholders, especially between software engineers and risk analysts, is of utmost importance to apply the proposed process successfully.

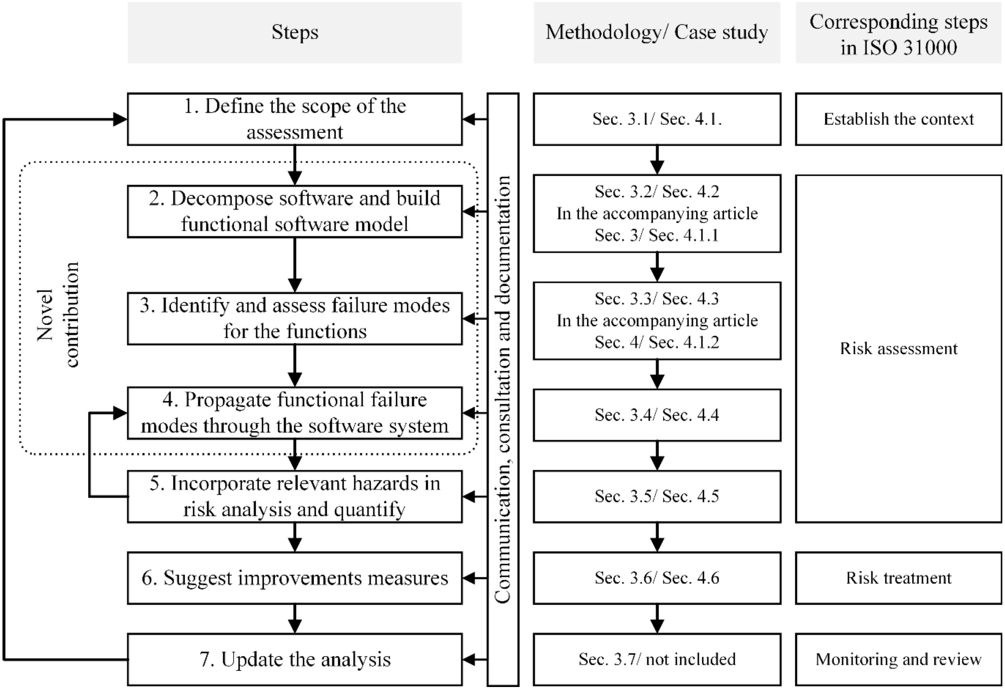

Steps
Methodology/ Case study
Corresponding steps in ISO 31000
1. Define the scope of the assessment
2. Decompose software and build functional software model
3. Identify and assess failure modes for the functions
4. Propagate functional failure modes through the software system
5. Incorporate relevant hazards in risk analysis and quantify
6. Suggest improvements measures
7. Update the analysis
Novel contribution
Communication, consultation and documentation
Sec. 3.1/ Sec. 4.1.
Sec. 3.2/ Sec. 4.2
In the accompanying article
Sec. 3/ Sec. 4.1.1
Sec. 3.3/ Sec. 4.3
In the accompanying article
Sec. 4/ Sec. 4.1.2
Sec. 3.4/ Sec. 4.4
Sec. 3.5/ Sec. 4.5
Sec. 3.6/ Sec. 4.6
Sec. 3.7/ not included
Establish the context
Risk assessment
Risk treatment
Monitoring and review

Figure 1 Steps in the proposed process to incorporate software failure in risk assessment and the corresponding steps in the ISO 31000 risk management framework [3]. Abbreviation: Sec. – Section.

3.1 Step 1: Define the Scope of the Assessment

The definition of the scope includes an overall description of the software, its purpose, application area, and operational context. Risk analysis is context specific, and the model should reflect this. The operational context describes which interactions the program has with its environment, such as other software programs, servers, humans, or sensors. Every interaction or output that is different from the expected interaction or output is a failure of the software. Only with the context, it is possible to analyze which failures will cause negative consequences.

The stage of the software in its life cycle determines the level of detail of the risk analysis. The level of detail of the risk analysis needs to be defined. Available documentation for the software, such as the software requirements specification (SRS) (according to IEEE 830 [19]), the system requirements specification, the software development documentation, or the verification and validation documentation, needs to be identified and used in the analysis process. Software engineers should be involved in the process and development of the functional software model to avoid ambiguity and increase understanding of the software system.

3.2 Step 2: Decompose Software and Build Functional Software Model

A functional decomposition of the software system is the first step toward building the functional software model. The functional decomposition and description of functions is necessary to collect and arrange the necessary information for the next steps. The functional analysis standard EN14514 [17] may assist in the decomposition. The accompanying article (Thieme et al. [1]) provides more information on functional decomposition and the description of the functions.

The functional decomposition is used to build the functional software model, which graphically represents the collected information. The functional software model visualizes the interactions between the functions and assists the analysts in maintaining an overview of the functions and their relationships. The connections between the functional elements are constructed according to the information on inputs, outputs, and associated conditions.

Figure 2 summarizes the symbols for building the model. The function descriptions and the behavior are associated with each of the functions. Two types of connectors are used in the functional software model. *Transfer of information* refers to the connection of functions through common data (i.e., the input and output). The second type, *functional dependency*, describes the influence of functions on other functions that are not related through the exchange of data. This could be functional calls or prerequisite functions. The *software boundary* is used as a visual cue to differentiate the external interfaces from the software functions.

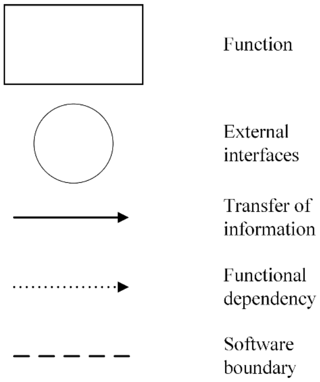


Figure 2 Modeling elements to represent the software functionality.

The information collected in the functional software model and the associated information on the functions assist in the analysis of the interaction failure modes (Step 3) and the analysis of the propagation behavior (Step 4). The description within the blocks needs to be coherent throughout the model to facilitate these steps.

3.3 Step 3: Identify and Assess Failure Modes for the Functions

This step is central to the proposed process since potential failure modes are identified for each function. These function-specific failure modes are propagated in the next step to analyze the effect of each individual failure mode.

The accompanying article [1] presents the failure mode taxonomy used in this article. There are four categories: functional, interaction, timing-related, and value-related failure modes. The failure mode taxonomy suits the functional view of software adopted in this article.

The analysts need to assess which failure modes are applicable to the software functions. Each identified failure mode needs to have a unique identifier to make it traceable in further analysis. Each failure mode should be described according to the chosen level of analysis. The analysis should consider the complete information to give meaning to the failure modes. Especially functional and non-functional requirements and constraints need to be included in these considerations.

3.4 Step 4: Propagate Functional Failure Modes through the Software System

The output and hence the effect of each failure mode on the external interfaces needs to be analyzed with respect to the overall system functionality and the context. The critical aspect in this step is how the failure modes interact with the external interface through the propagation behavior. The analysis needs to assign an effect in a meaningful manner to the propagated failures. The failure modes are propagated until all reachable interfaces are affected. The importance of considering failure propagation is explained in the accompanying article [1].

Generally, the propagation of the failure modes highly depends on the software functions and its overall function. The effect of control loops and reiterations within the software shall be considered. The propagation shall be reiterated at least once for loops, such that the effect of these will become visible. Additional iterations may be necessary. Fault detection and correction mechanisms need to be considered while analyzing the failure propagation behavior.

Table 2 summarizes the propagation behaviors of the failure modes through a software system. The first column summarizes the failure modes. The second column is labeled *refined failure mode*. Refined failure modes describe the failure mode in more detail and reflect a higher level of detail of the analysis. The third column describes the propagation behavior of the failure mode. The column *Ref.* describes the source from which the propagation behavior was derived. In this case, *1* refers to Wei [20] and *2* refers to the authors' identified propagation behavior.

Value-related failures affect subsequent functions by providing an incorrect value. The effect depends on the functionality and the process in the subsequent functions. In most cases, the value failure will lead to an incorrect value failure. Effectively, decisions and output to the external interfaces will be affected by these incorrect values and/or dependent function calls. The propagation and hence the overall effect on the external interfaces is highly dependent on the software purpose.

Functional failure modes mainly propagate similar to value-related failure modes. Propagation of interaction failure modes depends on the function process and interactions. Not calling or skipping functions will mostly propagate as the failure modes *no value* or *output provided too late*. In most cases, the failure modes related to external files will propagate as the *no value* failure mode.

For timing-related failures modes, three cases are differentiated [20]: no fault tolerance mechanisms with respect to timing (T1), watchdog timers or similar (T2), and failure recovery mechanisms with respect to timing (T3). In the case of T1, these failures will propagate directly through the software functions. In the case of T2, the software will either abort or exhibit a safe behavior. Safe behavior refers to a standard functional call or usage of a safe standard value. Moreover, in the case of a detected failure, T3 refers to software that will execute actions that will reduce the negative effects of the failure mode [20].

If data-rate failures are considered, then the design of the data transfer system becomes relevant [20]. In sporadic mode (SM), the data receiving function is activated by the available data. Data can be transferred in a passive mode (PM), and the data receiving software functions check all events and data available in the associated buffer. In active mode (AM), the buffer pushes out old data when it is full and the software function has yet not handled the data. A polling system specifically requests data as soon as the software function requires input [20].

Table 2 Propagation behaviors for each failure mode. Propagation behaviors for timing and value-related failure modes were adapted from Wei [20] (marked with 1 in the Ref. column). Other failure mode propagation mechanisms are based on the authors' assessment (marked with 2). Refined failure modes refer to special cases of failure modes.

Failure mode	Refined failure mode	Propagation behavior	Ref.
Function failure modes			
Omission of a function/missing operation		Propagates as incorrect value failure mode or no value failure mode.	2
Incorrect functionality		Propagates as incorrect value failure mode.	2
Additional functionality		Propagates as incorrect value failure mode (e.g., for outputs that shall not be manipulated). Can also propagate as output provided spuriously failure mode.	2
No voting		Propagates as no value failure mode.	2
Incorrect voting		Propagates as incorrect value failure mode (for the voting result).	2
Failure in failure handling		Detected failure are not handled, and the failure propagates as no value failure mode.	2
Interaction failure modes			
Diverted/incorrect functional call		Failure mode propagates in different ways, depending on the function (i.e., no value, incorrect value, or output provided spuriously).	2
No call of next function		The program stalls, propagates as failure modes: no value, or output not provided in time.	2
No priority for concurrent functions		Output is propagated with output provided too late failure mode.	2
Incorrect priority for concurrent functions		Output is propagated with output provided too late failure mode.	2
Communication protocol dependent failure modes		These failure modes include the generic failure modes and propagate accordingly.	2
Unexpected interaction with input-output boards		Propagates as output provided spuriously.	2
Failure of interaction with external files or databases	Wrong name	Propagates as no value failure mode.	2
	Invalid name/extension	Propagates as no value failure mode.	2
	File/ database does not exist	Propagates as no value failure mode.	2
	File/ database is open	Writing: propagates as no value failure mode Reading: might not propagate or propagates as no value failure mode.	2
	Wrong/invalid file format	Propagates as no value failure mode.	2
	File head contains error	Propagates as no value failure mode.	2
	File ending contains error	Propagates as no value failure mode.	2
	Wrong file length	Propagates as no value failure mode.	2
	File/database is empty	Propagates similar to too many elements in data array/structure.	2
	Wrong file/database contents	Propagates as no value failure mode.	2
Timing-related failure modes			

Failure mode	Refined failure mode	Propagation behavior	Ref.
Output provided	Too early	T1[1]: No output is registered, propagates as no value failure mode.	1
		T2[2]: Failure is detected, and the software aborts the operation.	1
		T3[3]: Failure is masked; the premature value is stored in a buffer and available for further operation.	1
	Too late	T1: Fault propagates as delayed output by the same time as the initial delay.	1
		T2: Delay is detected if it is longer than the programmed interval, software aborts operation.	1
		T3: If the delay is longer than the specified interval a standard value/behavior is used that is propagated as incorrect value failure mode.	1
	Spuriously	No output is registered, propagates as *no value* failure mode. Alternatively, a spurious action is triggered that will propagate as *too early* failure mode.	2
	Out of sequence	No output is registered, propagates as *no value* failure mode.	2
	Not in time	See output provided too late, where, for T1, the output is not provided in time.	2
Output rate failure	Too fast	SM[4]: Propagates as *too early* failure mode.	1
		AM[5] (drop new data) or PM[6], within affordable rate: Propagates as *too early* failure mode.	1
		AM (drop new data) or PM, faster than affordable rate: Buffer fills too fast, loss of data propagates as *incorrect value* failure mode. If buffer handles events, these events are lost and the system does not react accordingly.	1
		AM: push out old data: The output propagates as *incorrect value* failure mode, since the value that is assumed to be read is different from the assumed value.	1
	Too slow	PM: Output rate is the input rate. The *too slow* failure is propagated.	1
		AM: Old values stored in the buffer are used, propagates as *incorrect value* failure mode.	1
		PS: Output rate is the input rate. *Too slow* failure mode is propagated.	1
	Inconsistent	Propagates as *incorrect value* failure mode, pairing values from different times.	2
	Desynchronized	Propagates as *incorrect value* failure mode, taking the value from the synchronization delay.	1
Duration	Too long	Duration of a measurement: Output is propagated as *too high value* failure mode.	1
		Duration of detecting a presence: Signal is recognized multiple times, propagates as *too high failure*.	1
	Too short	Duration of a measurement: Output is propagated as *too low value* failure mode.	1
		Duration of detecting a presence: Signal is not recognized, program does not execute the command, propagates as *output not provided in time* failure mode.	1

[1] No failure detection mechanism with respect to timing
[2] Failure detection mechanism,
[3] Failure detection and recovery mechanisms,
[4] Sporadic mode.
[5] Active mode, PM – passive mode
[6] Passive mode

Failure mode	Refined failure mode	Propagation behavior	Ref.
Recurrent functions scheduled incorrectly		Propagates as *output provided spuriously or output not provided in time* failure modes.	2
Value-related failure modes			
No value		Either the next function waits for the value, propagating as *too lat*e failure mode, or a predefined value is used, propagating as *incorrect value*.	2
Incorrect value	Too high Too low Opposite/inverse value Value is 0 (zero)	The value failure propagates through the software, assuming that the value is correct. The value will lead to wrong computational results and this wrong information will be used during further evaluation. If the computed result falls out of the expected or allowable range, the value will propagate as *out of range* failure mode.	1, 2
Value out of range	Datatype allowable range	Value is adjusted to fit in the range and will propagate as *incorrect value* failure mode.	1
	Application allowable range	Value is adjusted to the closest allowable value of the range and propagates as *incorrect value* failure mode with this value.	1
Redundant/frozen value		Value propagates with the value as incorrect value.	2
Noisy value/precision error		Depending on magnitude, will lead to an *incorrect value* failure mode and propagate as such.	2
Value with wrong datatype		Depending on the type of the conversion, different propagation mechanisms were identified. The failure mode might not influence the value and be masked, leading to a loss of precision or incorrect value failure modes. If failure detection mechanisms can detect the failure, the operation will be aborted, and the software will continue as specified. For a detailed list of datatype errors, see Wei [20].	1
Non-numerical value	Not a number (NaN)	Corresponds to an undefined value conversion; hence, it will propagate according to the propagation mechanisms for value with wrong datatype.	2
	Infinite Negative infinite	Will propagate as *incorrect value out of range* failure mode.	2
Elements in a data array/structure	Too many	Elements come from different components. Error not propagated, additional input neglected.	1
		Elements come from one component, are read in fixed format, and are added to the end. Error not propagated, additional input neglected.	1
		Elements come from one component, are read in fixed format, and are inserted in the data array/structure. *Incorrect value* failure mode propagation from the element of insertion.	1
		Elements come from one component, are read in unfixed format, and are added at the end of the data array/structure. *Incorrect value* failure mode propagation of the last element.	1
	Too few	Elements come from one component, are read in unfixed format, and are inserted the data array/structure. *Incorrect value* failure mode propagation of the remaining elements.	1
		Elements come from different components. Propagates as *too late* failure mode.	1
		Elements come from one component. Propagation as *no value* failure mode.	1
	Data in wrong order	For the elements that are wrongly ordered the failure mode will propagate as *incorrect value* failure mode. Value with wrong datatype failure modes might also be relevant.	2
	Data in reversed order	The failure mode will propagate as *incorrect value* failure mode, with the correct reversed values. Value with wrong datatype failure modes might also be relevant.	2

Failure mode	Refined failure mode	Propagation behavior	Ref.
	Enumerated value incorrect	If the value lies within the range of the array/structure, it will be propagated as *incorrect value* failure mode. If falls out of the range it will lead to a program crash or will be handled by the failure detection mechanism.	2
Correct value is validated as incorrect		Correct values are rejected. Propagated as *too late*, or *output not provided in time* (c.f. timing failure modes).	2
Incorrect value is validated as correct		Incorrect value is propagated as *incorrect value* failure mode.	2
Data is not validated		Propagated as *too late*, or *output not provided in time* (c.f. timing failure modes or software aborts).	2

3.5 Step 5: Incorporate Relevant Hazards in Risk Analysis and Quantify

In this step, the analysis identifies and incorporates the safety-relevant undesired effects that were identified through the failure mode propagation. These are the direct results from the propagation analysis in Step 4. Identified relevant effects may be implemented in FTs, event sequence diagrams, ETs, or as nodes in Bayesian networks.

Steps 4 and 5 are closely connected. Some iterations may be necessary to identify the relevant failure effects on the external interfaces that need to be incorporated in the risk analysis. This is symbolized in Figure 1 with the arrow pointing from Step 4 to Step 5.

The software failure mode effect on the external interfaces needs to be viewed in the context of use with other technical sub-systems or operator actions [21, 22]. Human operator actions may lead to software failures, but they may also correct and recover the system from software failure.

In addition to failures in the software, failures might arise in the interfaces of the software [11]. This might be faulty measurements from sensors, incorrectly entered data from human operators, or incorrectly implemented database queries. Applying the failure mode propagation behavior may be used to analyze the effect of an interface failure on the software system and consequently on the other external interfaces. This is not discussed further here and is subject to further work.

Quantification could be derived through expert judgment or software reliability models. However, the quantification of the identified failure modes and the propagated failure effects on the external interfaces is out of scope of this article and will not be discussed further.

3.6 Step 6: Suggest Improvement Measures

Risk assessment is used to assess the risk level of an activity and propose mitigating measures in case of high levels of risk. Measures to improve the software system are (among others) to specify additional software functionality, redesign the software system, or specify additional safety and functional requirements for the software system.

The process for incorporating software in risk analysis should be used in the design phase of the software, such that necessary changes can be specified and implemented in an early stage of development.

The process may be applied to existing technological systems to estimate and include the risk contribution from the software system to the overall risk level. In contrast to hardware systems, software failure modes that are successfully removed from the software system will not reoccur under the same circumstances. Software updates that address identified failure modes and effects on the external interfaces need to be tested and verified.

The software system being analyzed should be tested, validated, and verified before it is used in operation to demonstrate compliance with the requirements. The results from the presented process to incorporate software in risk analysis could be used to generate test cases to ensure that critical failure modes will not occur. A formal software development process as laid out in ISO/IEC/IEEE 12207:2008 [23] and ISO/IEC/IEEE 15288:2015 [24] may assist in identifying the right risk mitigating measure.

3.7 Step 7: Update the Analysis

In accordance with the risk management standard ISO31000 [3], risk analyses need to be updated regularly. Several aspects might make it necessary to update the functional software model, the failure mode identification, and the associated risk analysis. These are change of context of use, change of interfaces, implementation of new functions, or implementation of failure identification and correction mechanisms.

3.8 Discussion

One important aspect for incorporating software in risk analysis of the proposed process is the propagation of identified functional software failure modes to identify their effects on external interfaces. The propagation behavior was partly adopted from the literature [20] and extended. Wei [20] defined propagation behavior for less failure modes than the accompanying article [1] covers. Therefore, this present article defines the propagation behavior for the failure modes from [1] that have not been covered previously.

The propagation behavior allows a consistent analysis of the software behavior if a functional software failure mode occurs. The purpose of the proposed process is to highlight possible weaknesses in the software system as a basis for improving the SRS and focus testing and verification efforts on critical aspects of the software system. This implies that a software project in an early phase should consider all failure modes and therefore will be aware of possible failure modes and associated propagated effects on the external interfaces.

Table 3 assesses the proposed process to incorporate software in risk analysis against the requirements that are presented in Table 1. All requirements are fulfilled except R5 and R7. Since the process is considering timing-related failure modes, R5 is only partly fulfilled. However, only through incorporation of the process in dynamic risk analysis is it truly possible to capture the full implications of timing-related failure modes in risk analysis [25].

Requirement 7, which is not fulfilled, addresses the quantification of the likelihood of software failure modes and their associated effects on the external interfaces. It is believed that it is possible to quantify software failure modes. A software tool may facilitate the process of analyzing the effect of propagating failure modes, their integration, and quantification in risk analysis.

Table 3 Assessment of the proposed process to incorporate software in risk analysis against the criteria from Table 1.

Requirement		Fulfillment	Comment
R1	Identify failure modes	Yes	Individual functional failure modes are identified for each function. The first part of this article identifies a comprehensive and coherent set of functional software failure modes.
R2	Identify possible failure causes	Yes	Failure causes can be found in the interfaces in the software itself or failure in the hardware support. The accompanying article outlines possible failure causes [1].
R3	Identify consequences of failure modes	Yes	Through consequent application of the failure mode propagation behavior, the consequences of software failure modes can be identified. The effects on the external interfaces can be integrated into risk analyses.
R4	Represent functional behavior	Yes	The functional behavior of the software system is explicitly modeled and represented through the functions.
R5	Represent temporal behavior	Partly	The temporal behavior is included in the model through timing constraints, requirements, and timing-related failure modes.
R6	Represent context of use	Yes	The context of use of the software is represented by including external interfaces in the functional software model, considering the overall requirements, and using context-specific failure modes for a certain situation.
R7	Quantify likelihood of consequences	No	The process for incorporation of software in risk analysis allows for quantification of the failure effects in risk models (e.g., FT and ET). However, the quantification process is not covered in this article.
R8	Be modular	Yes	The functional software model is modular through the functional decomposition. Each function is represented as its own module.
R9	Be scalable	Yes	The process for incorporating software in risk analysis is scalable. It can be used for large and small software systems. The interactions between the functions are known and hence can be modeled. The process can focus on different levels of detail and functional decomposition.
R10	Make use of all available information	Yes	The functional software model uses and reflects all the information that is collected in the SRS and other documentation.
R11	Be applicable throughout software life cycle	Yes	Through the scalability and modularity, the process can be applied at different stages of development. Especially in the operation phase, the modularity makes it easy to adapt the model to changes.

The other requirements are fulfilled. The requirements that are fulfilled and differentiate the proposed process to incorporate software in risk analysis from suggested methods and processes are, among others, R5, R6, R10, and R11. The difference from existing methods and processes will be discussed in more detail below.

The proposed process in this article allows identification of functional failure modes, failure consequences, and failure causes, which addresses R1 to R3. The process allows representing the context and functional behavior (R4 and R6). Failure modes are identified for the functional behavior. The effects of the functional failure modes may be integrated in risk analysis, thus integrating it in the context.

The proposed process is modular and scalable (R8 and R9), which originates from the functional approach. The functional approach also allows using the proposed process in different life cycle phases (R11). The process makes use of all available information, building the model and assessing failure modes based on that information.

Generally, the proposed process requires a good understanding of the software to be analyzed and the software development process. It is necessary that the risk analyst and software developers work together and develop a common understanding of both the software and risk analysis, such that ambiguities can be avoided.

The presented process is not the first to attempt to identify and incorporate software failures into risk analysis. Wei et al. [26] applied failure modes and identified their effects in a simulation environment. Wei et al. [26] only applied selected failure modes to some of the software functions. Their approach requires that the full software is available. However, not all the information that might be available from the software development process is incorporated. Hence, the approach by Wei et al. [26] does not completely fulfill the requirements R8, R10, and R11.

The presented process in this article differs significantly from a software failure mode and effect analysis (FMEA). In most cases, FMEA assesses the effect of a failure mode based on discussion and knowledge of the analysts, and not all available information is used (R10). In a FMEA, only the most critical or likely failure modes are included [27]. The FMEA alone does not allow for quantification of failure events for quantitative risk analysis (R4). Moreover, FMEA is most suitable for risk analysis in the design phases of a system [28] (R11).

The suggested process for incorporating software in risk analysis focuses on the software and its interactions with external interfaces and implementation of relevant failure events in risk analysis. This is different from other methods and processes, such as system-theoretic process analysis (STPA) [29-31] or the dynamic flowgraph method (DFM) [32-34], which focus on the identification of hazardous events. These methods do not address requirements focusing on quantitative assessment (R4). In addition, DFM does not provide mechanisms for identifying failure causes (R2).

4 Case Study

This section exemplifies the process to incorporate software in risk analysis on a software-based decision support tool with risk relevant implications. Each step of the proposed process will be addressed, except Step 7. A complete analysis of the software system would be too extensive. Hence, only selected aspects of the case study object will be presented in detail.

Steps 2 and 3 are briefly presented in the case study of the accompanying article [1]. More explanation is provided here to provide enough information for the failure mode propagation in Step 4.

4.1 Step 1: Define the Scope of the Assessment

Hegde et al. [35, 36] presented collision avoidance rules based on safety envelopes for an autonomous remotely operated vehicle (AROV). They implemented the set of traffic rules in a software tool to provide decision support in AROV operations, the underwater collision avoidance system (CAS). Since the software provides decision support with respect to the safe operation of the AROV, it is necessary to assure that the tool does not increase the level of risk.

The underwater CAS receives data through external interfaces and provides information for operational decision making. It is developed in an academic setting, not following the lifecycle processes of software, as laid out in ISO/IEC/IEEE 12207:2008 [23] or ISO/IEC/IEEE 15288:2015 [24]. The main developer is a co-author of this article and provided necessary information and input for the analysis.

4.1.1 Context for the analysis

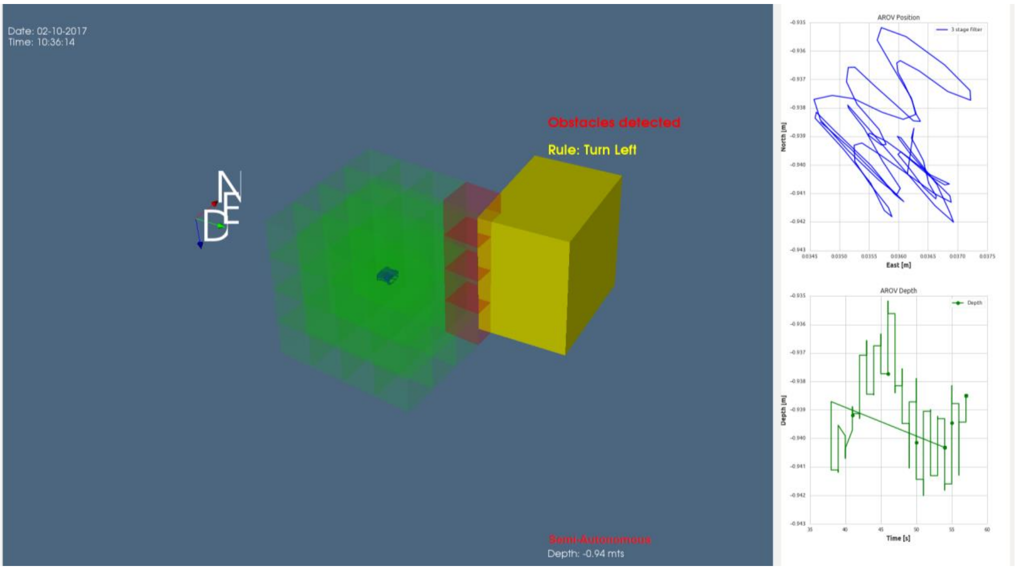
The underwater CAS by Hegde et al. [35, 36] has four aims: visualize the detection of static obstacles using safety envelopes, suggest a change of course based on safe traffic rules if an obstacle is detected, provide three dimensional (3D) orientation and position visualization, and visualize the traversed path in time and space.

The underwater CAS is designed for the operation of AROVs, which are unmanned underwater vehicles that operate mostly autonomously. The program has two main assumptions: (i) the size and position of all detected obstacles are known and (ii) the exact position of the AROV is known. The underwater CAS is programmed with the language Python 2.7. The user interface was created with Qt and was converted to python code. The renderer of the 3D model uses the Visualization Toolkit library. The plots are realized with the Matplotlib library.

The underwater CAS receives its input from an external interface. The Mission Oriented Operating Suite (MOOS) database provides position, attitude, and collision data. In addition, MOOS is a middleware developed to access the mission-related parameters [37]. The MOOS database collects and stores data produced by the AROV and associated software. The data can be requested from the AROV components that need parts of the data. The underwater CAS produces outputs. It sends requests to the MOOS database for position, orientation, and identified collision candidates, and it visualizes the 3D model, position plots, and status messages regarding recommended actions to the human operator via a screen.

For the analysis, a transit of the AROV from a subsea garage to a working site in an underwater oil and gas production facility is assumed. The AROV moves with velocity of 1.5 m/s. The distance from the center of the AROV to the outer envelope is 2.5 m. During the transit, the AROV passes a subsea structure.

The structure is detected within the outer safety envelope. The expected recommendation of the underwater CAS is to execute an evasive maneuver to the left of the structure to keep a safe distance from the obstacle. The situation of the analysis is rendered in Figure 3. The AROV follows pre-programmed waypoints. If the AROV detects obstacles, the underwater CAS will warn the human operator. The underwater CAS will suggest an evasive maneuver, and the human operator needs to implement a route. The human operator could also take direct control of the AROV using the control joysticks. Although the underwater CAS is a conceptual development, it is assumed that it is part of the human-machine interface of the human operator with the AROV and hence assist in the operation.



Date: 02-10-2017
Time: 10:36:14

Obstacles detected

Rule: Turn Left

Semi-Autonomous
Depth: -0.94 mts

Figure 3 Situation visualization for the case study; the plots on the right hand side are a visual example, not representing the current situation.

The implementation of the safety envelopes in the MOOS database and the AROV control has been verified and demonstrated [35, 36]. The traffic rules are assumed to be implemented correctly in the underwater CAS. It has been verified that the MOOS database gives expected datatypes and outputs in the right format.

4.1.2 Aim of the risk assessment

The analysis focuses on how the underwater CAS could contribute to a collision with the subsea structure that the AROV shall pass. Based on the above-described situation, the possible effects of the software on the external interfaces are analyzed with the failure modes and the propagation behavior. The results of the analysis shall be implemented in qualitative FTs to analyze the effect on the overall operation.

The application of the process shall give input to potential mitigation measures and shall help to improve the software during the next update. Other mitigating measures may be to adapt the system architecture. It shall also identify additional requirements or functionalities, which are necessary to avoid or mitigate the effect of possible failures.

4.2 Step 2: Decompose Software and Build Functional Software Model

The software decomposition can be found in the accompanying article [1]. Five functions were identified: initialize underwater CAS (F1), obtain data (F2), determine suggested action (F3), prepare render information (F4), and display information (F5).

In the first function, *initialize underwater CAS*, the program starts, establishes a connection to the database, and sets up the window for visualizing the data. In F2, *obtain data*, the software polls for the necessary information that the underwater CAS uses in the subsequent functions. The underwater CAS shall poll data from the MOOS database with a frequency of 2 Hz. The function is detailed in the accompanying article [1] and shall further serve as an example for the process in this article.

In F3, *determine suggested action*, information on the collision candidates and their positions is used to determine which actions are necessary to avoid a collision and stay at a safe distance. In F4, *prepare render information*, this information and the information on the collision candidates is used to highlight the corresponding safety envelope elements and display the recommendation. In addition, the 3D model is rendered according to the orientation of the AROV to give the human operator an overview of the situation.

The last function, *display information*, updates the plots for the position and the 3D model. This information is sent to the user screen, where the human operator will see the information and use it as aid for operating and monitoring the AROV.

Figure 4 presents the functional software model for the underwater CAS. It was developed from the functional decomposition and the description of the functions. All identified interfaces have been included. The program execution loop is represented through the broken line from F5 to F2. The diagram supports the analyses of failure modes and failure effect propagation in the next two steps. It illustrates the connection of the functions, the flow of information, and the dependency of functions. Each line is labeled with the associated output. These are described in Table 4. They represent the information that was summarized above.

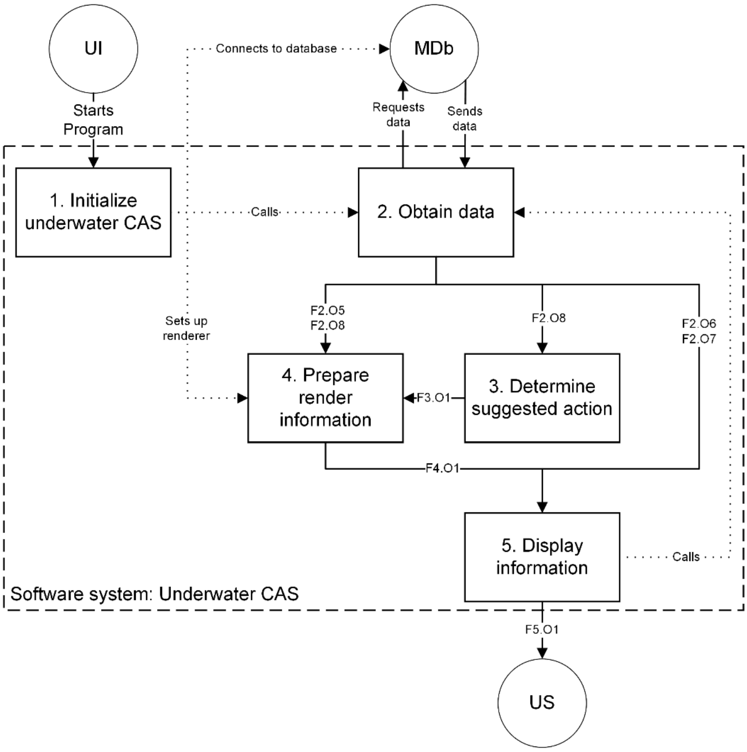


Figure 4 Functional software model of the underwater collision avoidance system software. Abbreviations: UI – User input; MDb – MOOS Database; US - User screen, description of the outputs can be found in Table 7.

Table 4 Description of the outputs of the functions of the underwater CAS, found in Figure 7.

Abbreviation	Name	Description
F2.O5	AROV orientation	Vehicle orientation in roll, pitch, and heading of the AROV.
F2.O6	AROV operational mode	Mode of operation of the AROV (i.e., remote control, semi-autonomous, autonomous).
F2.O7	AROV position	Local position of the AROV with respect to a local reference coordinate system, described in the north, east, and down reference frame.
F2.O8	Information on identified collision candidates	Information on objects that were identified as falling within the safety envelopes of the underwater CAS.
F3.O1	Suggested action	Suggested action to the AROV operator based on the current context.
F4.O1	Render information	Information necessary to update the renderer.
F5.O1	Screen information	Visualized information containing the render model, suggested action and position plot.

4.3 Step 3: Identify and Assess Failure Modes for the Functions

As mentioned in the previous section, F2, *obtain data*, is used as the case study object. The accompanying article [1] identified 36 failure modes for that function. The set of identified failure modes is incomplete. It focuses on demonstrating how most of the generic failure modes can be applied to the software function. The identified failure modes can be found in the first two columns of Table 5 in the next section. The failure mode identification will not be explained further here.

4.4 Step 4: Propagate Functional Failure Modes through the Software System

Table 5 summarizes the effects of applying the failure mode propagation behavior to the identified functional failure modes of F2. For the propagation of the failure modes, the information collected in Figure 4 is used. Information on the affected functions can be read directly from the functional software model.

In general, functions that are assessed with *no effect* are not influenced by the propagated failure mode. *No information updated or displayed* in the column *effect on user screen* can be interpreted as a crash or a hanging of the underwater CAS. The human operator will not receive any information. Some selected examples shall clarify the analysis process and provide additionally needed information in the following paragraphs.

The failure mode FM4, *incorrect functionality of storing values in the corresponding variables*, *making them unavailable,* will result in *no output* to the subsequent functions. These will not be able to produce their required output due to the missing data. Therefore, the user screen will not be updated, or any information displayed.

In FM8, *no function call to F3*, the software execution is affected in such a way that F4 will be executed directly. That means that the render information is prepared and sent further to function F5. In this case, F5 will prepare the display data without the suggested action since it was not determined. Hence, the user screen will show all information correctly, except the suggested action.

With respect to timing-related failure modes, two examples will be further explained. *Output provided too late (500 ms): request for AROV orientation* (FM14), which is a delay in the execution of the functions that succeed F2, occurs. The program will periodically run the functions in the specified order. The human operator will experience the delay since the screen is not updated in real time but with the delay of 500 ms.

Table 5 Propagation behavior applied to the failure modes identified for Function 2 obtain data. The outputs are described in Table 3.

Failure modes		Propagation through F3	Propagation through F4	Propagation through F5	Effect on user screen
Function failure modes					
FM1	*Omission* of "Obtain data", which is not executed.	No F3.O1	No F4.O1	No F5.O1	No information displayed or updated
FM2	*Omission* of requesting data, which means that data is not requested.	No F3.O1	No F4.O1	No F5.O1	No information displayed or updated
FM3	*Omission* of converting MDb.O1 to AROV orientation data, which means that the orientation is note executed.	No effect	F4.O1 with wrong orientation	F5.O1 prepared with wrong orientation	Rendered model displayed with wrong orientation
FM4	Incorrect functionality of storing values in the corresponding variables, making them unavailable	No F3.O1	No F4.O1	No F5.O1	No information displayed or updated
FM5	Additional functionality while converting AROV orientation (e.g., conversion of AROV position)	No effect	No effect	F5.O1 prepared with wrong position	Plots displayed with wrong position
FM6	Failure in failure handling, no detection that no value has been received	No F3.O1	No F4.O1	No F5.O1	No information displayed or updated
Interaction failure modes					
FM7	Incorrect function call, calling F4, skipping F3	F3 not executed	Updated without F3.O1	F5.O1 prepared without suggested action	Information displayed without suggested action
FM8	No function call to F3	F3 not executed	Updated without F3.O1	F5.O1 prepared without suggested action	Information displayed without suggested action
FM9	Incorrect priority for functions, call function F4 before F3	Executed after the renderer is updated	Updated with F3.O1 with old information	F5.O1 prepared with old suggested action	Information displayed with old suggested action
FM10	Unable to request information from the database (communication protocol-dependent failure)	No F3.O1	No F4.O1	No F5.O1	No information displayed or updated
FM11	Request with wrong variable name to the database for AROV position	No effect	No effect	F5.O1 prepared without position information	No update of AROV position plot
Timing-related failure modes					
FM12	Output provided too early: Request for AROV orientation	Output provided too early	Output provided too early	F5.O1 provided too early	Information on screen is updated earlier than required
FM13	Output provided too late: Request for AROV orientation	Output provided too late	Output provided too late	F5.O1 provided too late	Information on screen is updated later than required
FM14	Output provided too late (500 ms): request for AROV orientation	Output provided 500 ms late	Output provided 500 ms late	F5.O1 provided 500 ms late	Information on screen is updated 500 ms later than required

Failure modes		Propagation through F3	Propagation through F4	Propagation through F5	Effect on user screen
FM15	Output provided spuriously: AROV operational mode	No effect	No effect	No effect	Information on screen is incorrectly updated
FM16	Output provided out of sequence: Information on identified collision candidates provided before AROV position	No effect	No effect	No effect	Information on screen is incorrectly updated earlier than required
FM17	Output not provided in time: Information on identified collision candidates	Output provided too late (delay determined by delay in F2)	Output provided too late (delay determined by delay in F2)	F5.O1 provided too late (delay determined by delay in F2)	Information on screen is updated later than required (delay determined by delay in F2)
FM18	Output rate too fast: Requests to database send too fast (within affordable rate)	Output provided too early	Output provided too early	F5.O1 provided too early	Information on screen is updated earlier than required
	Output rate too fast: Requests to database send too fast (out of affordable rate, data dropped)	No F3.O1	No F4.O1	No F5.O1	No information displayed or updated
FM19	Output rate too slow: Requests to database send too slow	Output provided too late	Output provided too late	F5.O1 provided too late	Information on screen is updated later than required
FM20	Inconsistent rate for requests	Output provided inconsistently in time	Output provided inconsistently in time	F5.O1 provided inconsistently in time	Information on screen is updated inconsistently
Value-related failure modes					
FM21	No value for AROV position	No effect	No effect	F5.O1 is not containing an position update	No update of AROV position plot
FM22	Incorrect value for AROV position (not further defined)	No effect	No effect	F5.O1 is not containing the right position	Plots displayed with wrong position
FM23	Incorrect value, too high for AROV operational mode = 2	No effect	No effect	F5.O1 contains wrong operational mode “autonomous”	Display of information that AROV is in autonomous mode
FM24	Incorrect value, too low for AROV operational mode = 0	No effect	No effect	F5.O1contains wrong operational mode “manual”	Display of information that AROV is in manual mode
FM25	Incorrect value, too high, AROV orientation [0, 0, -15]	No effect	Render model prepared with wrong heading orientation (- 15°)	F5.O1 prepared with - 15° wrong heading	Render model displayed with -15° wrong heading
FM26	Incorrect value, too high, AROV orientation [0, 0, -30]	No effect	Render model prepared with wrong heading orientation (- 30°)	F5.O1 prepared with - 30° wrong heading	Render model displayed with -30° wrong heading

Failure modes		Propagation through F3	Propagation through F4	Propagation through F5	Effect on user screen
FM27	Incorrect value, zero for AROV position [0,0,0]	No effect	No effect	F5.O1 prepared with position displayed as origin of the local coordinate system	AROV position displayed as origin of the local coordinate system
FM28	Value out of application allowable range for Information on identified collision candidates includes the value 68	No F3.O1	No F4.O1	No F5.O1	No information displayed or updated
FM29	Value out of datatype range for AROV operational mode = 2,147,483,648	No effect	No effect	Operational mode adjusted to closest allowable value (0), F5.O1 prepared with wrong operational mode“	Display of information that AROV is in manual mode
FM30	Frozen value for Information on identified collision candidates (no collision candidates detected)	F3.O1 = no suggested action	F4.O1 without highlighted envelope elements	F5.O1 prepared without suggested action and without highlighted envelope elements	Display of information that no action is needed, and no collision candidates were detected
FM31	Imprecise value for AROV position varying more than 1 m	No effect	No effect	F5.O1 prepared with imprecise position (+/- 1m)	AROV position imprecisely displayed (+/- 1m)
FM32	Wrong datatype for AROV operational mode, string instead of int	No effect	No effect	No effect, as long as the value is 1	No effect
FM33	Too many (65) elements, in information on identified collision candidates	No effect, as long as the value is within the range	No effect	No effect	No effect
FM34	Too few elements (two elements instead of three) in AROV orientation	No effect	No F4.O1	No F5.O1	No information displayed or updated
FM35	Data in wrong order in AROV position [z, x, y] instead of [x, y, z]	No effect	No effect	F5.O1 prepared with wrong position	Plots displayed with wrong position
FM36	Incorrect value (no value) for F2.O5-F2.O8 is validated as correct and is output	No F3.O1	No F4.O1	No F5.O1	No information displayed or updated

Abbreviation: FM – Failure mode, MDb – MOOS database, Functions: F2 – Obtain data, F3 – Determine suggested action, F4 – Prepare render information, F5 – Display information

In FM16, *output provided out of sequence: information on identified collision candidates provided before AROV position,* there is *no effect* on the output. The information is stored in dedicated variables. Unless the information is stored to the wrong variables, it will not affect the output to the external interfaces.

The failure modes FM23 and FM24, *incorrect value, too high, AROV orientation [0, 0, -15*]/ *[0, 0, -30]*, respectively, are a special demonstration of how similar failure modes might affect the risk level. In this case, the heading of the vehicle is shifted in the failure mode by -15° and -30°, respectively. This failure will affect the model of the AROV being displayed with a wrong heading. Incorrect orientation display might have different implications for the human operator.

Regarding FM28, *value out of application allowable range for information on identified collision candidates includes the value 68*, the failure mode will propagate as no output. The output will lead to no output in F3 since the value cannot be interpreted. No mechanisms are in place to check whether the value falls in the range. The *no output* failure mode will propagate to the screen, and the human operator will experience it as a hanging or crashing of the program.

Similarly, FM 34, *too few elements, (two elements instead of three), in AROV orientation*, will lead to no output in Function 4. Function 3 is not affected since it does not use the information in the output *AROV orientation*. In Function 4, the program will read from the array, which only has two elements and not the expected three elements. When trying to read the third element, the function will not be able to do so and cannot produce an output. The human operator again will experience this as hanging or crash.

4.5 Step 5: Incorporate Relevant Hazards in Risk Analysis and Quantify

This section shall demonstrate how the identified effects on the external interfaces and the safety-relevant effects can be implemented in the risk analysis. For that purpose, a fault tree analysis (FTA) was conducted. The top event for the FT is *collision with subsea structure during transit*. It incorporates human- and software-related events. The developed FT covers only part of the complete risk analysis. The FT has not been quantified since this is out of the scope of this article.

Figure 5 and Figure 6 present the developed FT, which is split into two parts for better readability. The effects on the interfaces from the propagated failure modes that relate to the display of wrong information are presented in Figure 5.

The effects on the interfaces from the propagated failure modes that relate to the omission of displaying information can be found in Figure 6. Examples are *no information displayed or updated* or *no update of AROV position plot*. These events are only relevant if the human operator needs to rely heavily on the underwater CAS, due to visibility or technical conditions, and if the human operator decides to continue the mission, despite the degraded performance of the underwater CAS. Two events in the FTs are undeveloped, these relate to the failure in the control system and human operator failure during waypoint planning or implementation.

The main part of the FT, Figure 5, includes some of the events that relate to a wrong display of information or delayed output of information. The AND-Gate 3, for example, contains events in which the information is provided too late with respect to the requirements. However, it might be possible that the human operator can take action beforehand or that the human operator can react and avoid a collision. Effects of propagated functional software failure modes that were included are *information on screen is updated 500 ms later than required*, *information on screen is updated later than required,* and *information on screen is updated inconsistently*.

Another group of effects of propagated functional software failure modes are those that relate to wrong information being displayed, such as position in AND-Gate 4, heading in AND-Gate 2, and *AROV operational mode being displayed as autonomous operation* in AND-Gate 5. Most of the events that will lead to a collision require the human operator to be fully trusting the information provided by the software, while not using other available information.

Not all of the identified effects of propagated functional software failure modes are relevant for the context. Hence, they were not included in the FTs. For example, *information on screen is updated earlier than required* does not influence the risk in relation to a collision. On the contrary, the earlier information is available and updated (an increased update frequency is implied) the better it is for the human operator.

Similarly, *display of information that AROV is in manual mode* was not included since the human operator will act, in this case. This is disregarding the possibility that the human operator will not take action due to other reasons. Such an event could be potentially found in the undeveloped event *operator failure during waypoint planning or implementation*.

The event *render model displayed with -15° wrong heading* was not included in the FT, since it is a rather limited change of heading and it falls in the normal variation of the AROV heading (e.g., to compensate for external disturbances). A deviation by more than that, in this case -30°, is assumed significant, such that the human operator will take action, in this case, one that may lead to a collision.

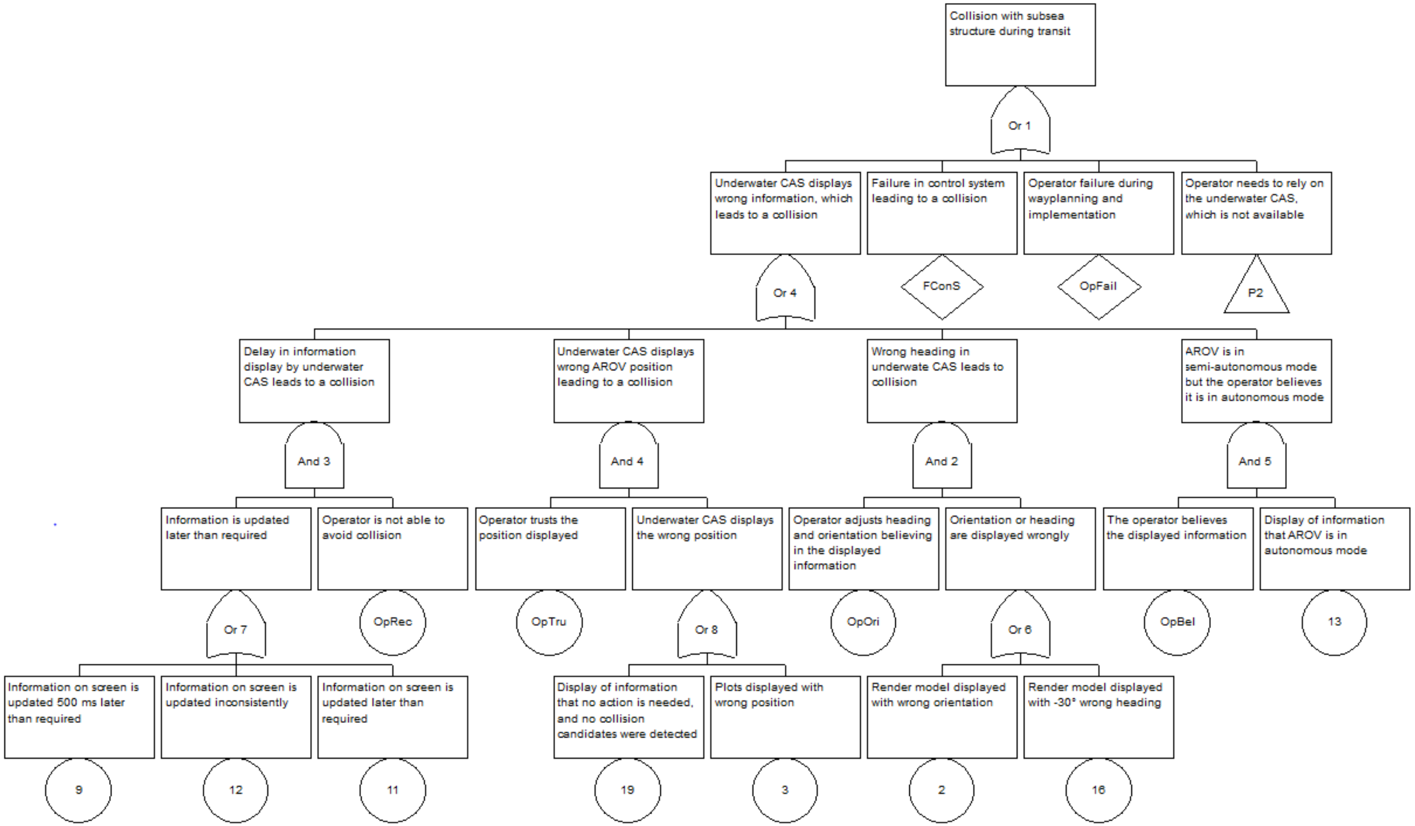

Collision with subsea structure during transit
Or 1
Underwater CAS displays wrong information, which leads to a collision
Failure in control system leading to a collision
Operator failure during wayplanning and implementation
Operator needs to rely on the underwater CAS, which is not available
Or 4
FConS
OpFail
P2
Delay in information display by underwater CAS leads to a collision
Underwater CAS displays wrong AROV position leading to a collision
Wrong heading in underwate CAS leads to collision
AROV is in semi-autonomous mode but the operator believes it is in autonomous mode
And 3
And 4
And 2
And 5
Information is updated later than required
Operator is not able to avoid collision
Operator trusts the position displayed
Underwater CAS displays the wrong position
Operator adjusts heading and orientation believing in the displayed information
Orientation or heading are displayed wrongly
The operator believes the displayed information
Display of information that AROV is in autonomous mode
Or 7
OpRec
OpTru
Or 8
OpOri
Or 6
OpBel
13
Information on screen is updated 500 ms later than required
Information on screen is updated inconsistently
Information on screen is updated later than required
Display of information that no action is needed, and no collision candidates were detected
Plots displayed with wrong position
Render model displayed with wrong orientation
Render model displayed with -30° wrong heading
9
12
11
19
3
2
16

Figure 5 Main fault tree with the top event *collision with subsea structure during transit*. The fault tree was developed with the effects from the propagated functional software modes.

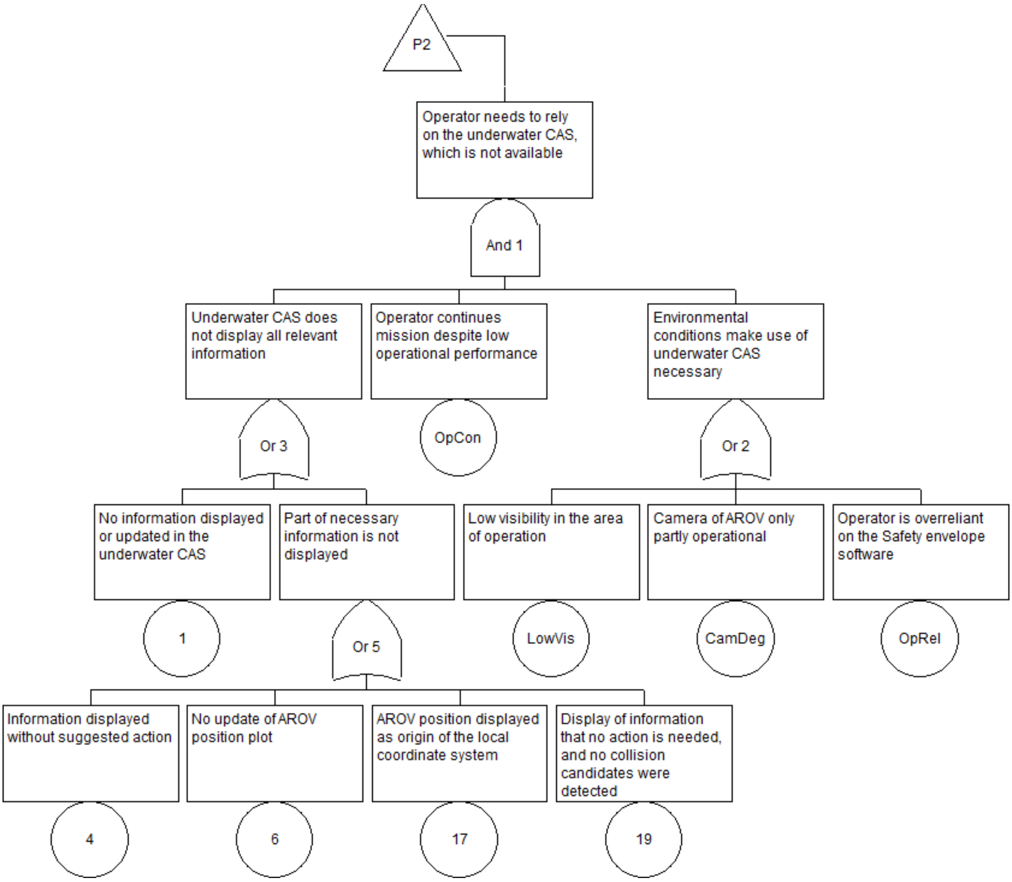

P2
Operator needs to rely on the underwater CAS, which is not available
And 1
Underwater CAS does not display all relevant information
Operator continues mission despite low operational performance
Environmental conditions make use of underwater CAS necessary
Or 3
OpCon
Or 2
No information displayed or updated in the underwater CAS
Part of necessary information is not displayed
Low visibility in the area of operation
Camera of AROV only partly operational
Operator is overreliant on the Safety envelope software
1
Or 5
LowVis
CamDeg
OpRel
Information displayed without suggested action
No update of AROV position plot
AROV position displayed as origin of the local coordinate system
Display of information that no action is needed, and no collision candidades were detected
4
6
17
19

Figure 6 Sub-fault tree for the transfer gate P2 of the fault tree *collision with subsea structure during transit*.

4.6 Step 6: Suggest Improvement Measures

Most of the failure modes and their propagation effects on the interfaces of the underwater CAS that were identified could be prevented by verifying that the data received is in the correct format and expected datatype. Several failures that may lead to a crash or hanging of the software can be avoided. In the current version of the program, no timing watchdogs or similar are implemented to ensure that the software will abort after a time without output. By defining such requirements and accordingly implementing them, hanging of the program can be detected and prevented.

In general, the underwater CAS was missing an implemented failure message system to the human operators. This should be implemented to assist the human operators in failure detection and solutions. Since the case study only covers a limited set of failure modes, no more improvement measures will be discussed.

4.7 Discussion

The case study was chosen due to its relevance for safe operation of an AROV and the potential for software improvement. Almost all the failure modes can be applied to the case study; hence, it is well suited for demonstration. Function 2 of the underwater CAS is described in detail. The analysis of other functions of the underwater CAS may be carried out similarly. The identification and propagation of software failure modes has been demonstrated. Only a few timing requirements are defined; therefore, only a few aspects of the timing-related failure modes could be demonstrated.

The example demonstrates that the effects of propagated failure modes on the external interfaces can be implemented in a risk analysis, in this case an FTA. The presented FTA uses a simplified FT, neglecting failures that might arise independently of the analyzed software. In a full risk analysis, these events may need to be considered. For example, the control system of the AROV should be analyzed with the proposed process.

Results from the case study show that software functional requirements and fault detection features can be identified to improve the software. This is addressed in the case study in Section 4.6. Analyzing the other functions and Function 2 completely could potentially identify additional relevant effects on the external interfaces, which should be implemented in the risk analysis. Consequently, this will lead to more specific recommendations for improvement of the software.

Some challenges are associated with the application of the proposed process to the underwater CAS. The software is developed in an academic setting, which does not apply a formal development process, as it may be used in the industry. However, it is believed that the example is representative for safety-relevant and related software systems and the risk assessment of these. The analyzed software is an important support system for the operation of AROV and might be implemented in future human-machine interfaces for AROV.

The proposed process is time intensive; thus, only one of the five functions of the underwater CAS was analyzed. Analyzing more complex software systems will be time-consuming. However, it will benefit the software being analyzed by deriving a comprehensive list of functional failure modes and their associated effects on external interfaces. Hence, an automated software tool should be developed and used to aid in the process.

5 Conclusion

This article presents a process for incorporating software failures in risk analysis, analyzing the effects of propagating functional failure modes on external interfaces, and incorporating these into the risk analysis. The process provides a systematic way to analyze the effects of functional failure modes on the software output and associated external interfaces.

The identified effects can be implemented in risk analysis and incorporated with human operator and hardware–related failure events. The process applies the propagation behavior of software functional failure modes. This is an advantage over the current methods for incorporating software in risk analyses since a structured process is applied.

Eleven requirements were developed to assess the process for incorporating software in a risk analysis. The proposed process fulfills these requirements, except for two. The proposed process does not fully capture the dynamics of the software with respect to the context; a dynamic risk analysis is required. The proposed process does not provide an approach to quantify the likelihood of the identified effects of propagated functional software failure modes on the external interfaces.

Relevant software failure effects are context specific and can be implemented directly in a risk analysis, via such methods as FTs, ETs, or event sequence diagrams. The case study in this article shows how such a venture could be conducted. It is believed that the proposed process can assist in identifying a cohesive set of software failure effects on its external interfaces of safety critical software and therefore improve the safety performance of the overall system.

The proposed process may be applied in the development phase of the software. It may aid in highlighting necessary measures to improve the software and make it safe before the software is written. The process may be applied to existing software systems, which makes it possible to improve existing software systems through updates and changes.

In the future, the process should be applied to more complex technical systems, such as autonomous ships, to demonstrate its applicability and feasibility. Future work should also incorporate the software failure effects on the external interfaces with human and organizational factors and the complete hardware system.

The process is time-consuming, and large software systems may be difficult to trace for the analysts. Hence, a software tool should be developed that aids in the process. The analysts shall focus on building the functional software model and implementing the relevant effects on the external interfaces of propagated software failure modes through relevant risk scenarios. This would be done of course with typical logic models to facilitate the overall process.

Regarding the implementation of the identified propagation effects in risk analysis, the proposed process does not contain an analysis of the effects of failures in the external interfaces on the software system. Including these propagation behaviors will improve the incorporation of mutual dependencies between software users, hardware, and software.

A second challenge lies in the quantification of the likelihood of failure modes and their effects on the external interfaces for risk analysis. Further investigation is needed to identify a suitable quantification method.

Acknowledgements

C. A. Thieme and I. B. Utne appreciate the support of the Research Council of Norway through the Centres of Excellence funding scheme, Project number 223254 – NTNU AMOS.

J. Hegde, has been supported by the Research Council of Norway, Statoil and TechnipFMC through the research project Next Generation Subsea Inspection, Maintenance and Repair Operations, 234108/E30 and associated with AMOS 223254.

C. A. Thieme acknowledges the financial support from Norges tekniske høgskoles fond, who supported this research with a scholarship for a research exchange with the B. John Garrick Institute for the Risk Sciences at the University of California in Los Angeles.

References

[1] Thieme CA, Mosleh A, Utne IB, Hegde J. Incorporating Software Failure in Risk Analysis – Part 1: Software Functional Failure Mode Classification. Submitted for review to Reliabilty Engineering and System Safety. Submitted: pp.

[2] OECD. Failure Modes Taxonomy for Reliability Assessment of Digital I&C Systems for Pra. Paris: Nuclear Energy Agency; 2014. pp. 135.

[3] ISO. Iso 31000 Risk Management - Principles and Guidelines. Geneva, Switzerland: International Organization for Standardization 2009.

[4] Marr B. The Future of the Transport Industry - Iot, Big Data, Ai and Autonomous Vehicles. 2017; https://www.forbes.com/sites/bernardmarr/2017/11/06/the-future-of-the-transport-industry-iot-big-data-ai-and-autonomous-vehicles/#2b854d791137; Accessed: 21.02.2018

[5] Kongsberg Maritime. Yara and Kongsberg Enter into Partnership to Build World's First Autonomous and Zero Emissions Ship. 2017; https://www.km.kongsberg.com/ks/web/nokbg0238.nsf/AllWeb/98A8C576AEFC85AFC125811A0037F6C4?OpenDocument; Accessed: 27.07.2017

[6] MUNIN. Maritime Unmanned Navigation through Intelligence in Networks. 2012; http://www.unmanned-ship.org/munin/ Accessed 23.07.2016; 23.07.

[7] Advanced Autonomous Waterborne Applications. Remote and Autonomous Ships - the Next Steps. In: Laurinen M, editor. Adavanced Autonomous Waterborne Applications London, UK 2016. pp. 88.

[8] Danish Maritime Authority. Analysis of Regulatory Barriers to the Use of Autonomous Ships. Regulatory scoping exercise for the use of maritime autonomous surface ships (MASS). Danmark: Maritime Safety Committee, Danish Maritime Authority (DMA); 2018. pp. 143.

[9] Huntsberger T, Woodward G. Intelligent Autonomy for Unmanned Surface and Underwater Vehicles. Oceans 2011. Kona, HI: IEEE; 2011.

[10] Mosleh A. Pra: A Perspective on Strengths, Current Limitations, and Possible Improvements. Nuclear Engineering and Technology. 2014;46: pp. 1-10.
[11] Ozarin NW. The Role of Software Failure Modes and Effects Analysis for Interfaces in Safety- and Mission-Critical Systems. 2008 IEEE International Systems Conference Proceedings, SysCon 2008, April 7, 2008 - April 10, 2008. Montreal, QC, Canada: Inst. of Elec. and Elec. Eng. Computer Society; 2008. pp. p 200-207.
[12] Ozarin NW. Applying Software Failure Modes and Effects Analysis to Interfaces. Annual Reliability and Maintainability Symposium 2009. pp. 533-538.
[13] Garrett CJ, Apostolakis G. Context in the Risk Assessment of Digital Systems. Risk Analysis. 1999;19: pp. 23-32.
[14] Chu T-L, Martinez-Guridi G, Yue M, Samanta P, Vinod G, Lehner J. Workshop on Philosophical Basis for Incorporating Software Failures in a Probabilistic Risk Assessment. Digital System Software PRA. Brookhaven National Laboratory; 2009. pp. 188 pages.
[15] Hewett R, Seker R. A Risk Assessment Model of Embedded Software Systems. 2005 29th Annual IEEE/NASA Software Engineering Workshop, SEW'05, April 6, 2005 - April 7, 2005. Greenbelt, MD, USA: Institute of Electrical and Electronics Engineers Computer Society; 2005. pp. 142-149.
[16] Kaplan S, Garrick BJ. On the Quantitative Definition of Risk. Risk Analysis. 1981;1: pp. 11-27.
[17] EN. Ns-En14514: Space Engineering Standards - Functional Analysis. Brussels, Belgium: European Committee for Standardization; 2004.
[18] IEC/ISO. Iec/Iso 31010: Risk Management - Risk Assessment Techniques. Geneva, Switzerland: International Organization for Standardization , International Electrotechnical Commision; 2009. pp. pp. 96.
[19] IEEE. Ieee 830: Recommended Practice for Software Requirements Specification. New York, NY, USA: IEEE; 2009.
[20] Wei YY. A Study of Software Input Failure Propagation Mechanisms. College Park, MD: University of Maryland; 2006.
[21] Ozarin NW. Bridging Software and Hardware Fmea in Complex Systems. 2013 Proceedings Annual Reliability and Maintainability Symposium (RAMS) 2013. pp. 1-6.
[22] Lindholm C, Notander JP, Höst M. A Case Study on Software Risk Analysis and Planning in Medical Device Development. Software Quality Journal. 2014;Vol. 22: pp. 469-497.
[23] ISO/IEC/IEEE. Iso/Iec/Ieee 15288: Systems and Software Engineering - System Life Cycle Processes. Geneva, Switzerland: International Organization for Standardization , International Electrotechnical Commission, Institute of Electrical and Electronics Engineers; 2015. pp. pp. 118.
[24] ISO/IEC/IEEE. Iso/Iec/Ieee12207:Systems and Software Engineering - Software Life Cycle Processes. Geneva, CH; New York, NY, USA: International Organization for Standardization , International Electrotechnical Commission, Institute of Electrical and Electronics Engineers; 2008. pp. pp.144.
[25] Zhu D, Mosleh A, Smidts C. A Framework to Integrate Software Behavior into Dynamic Probabilistic Risk Assessment. Reliability Engineering & System Safety. 2007;92: pp. 1733-1755.
[26] Wei YY, Rodriguez M, Smidts CS. Probabilistic Risk Assessment Framework for Software Propagation Analysis of Failures. Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability. 2010;Vol. 224: pp. 113-135.
[27] EN. Space Product Assurance - Failure Modes, Effects (and Criticality) Analysis (Fmea/Fmeca). Brussels, Belgium: European Committee for Electrotechnical Standardization; 2014.
[28] IEC EN. En Iec 60812: Analysis Techniques for System Reliability – Procedure for Failure Mode and Effects Analysis (Fmea). Brussels, Belgium: International Electrotechnical Commission, European Committee for Electrotechnical Standardization; 2006.
[29] Leveson NG, Fleming CH, Spencer M, Thomas J, Wilkinson C. Safety Assessment of Complex, Software-Intensive Systems. SAE International Journal of Aerospace. 2012;5: pp. 233-244.
[30] Abdulkhaleq A, Wagner S, Leveson N. A Comprehensive Safety Engineering Approach for Software-Intensive Systems Based on Stpa. Procedia Engineering. 2015. pp. 2-11.
[31] Abdulkhaleq A, Wagner S. Integrated Safety Analysis Using Systems-Theoretic Process Analysis and Software Model Checking. Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics) 2015. pp. 121-134.
[32] Garrett CJ, Guarro SB, Apostolakis GE. The Dynamic Flowgraph Methodology for Assessing the Dependability of Embedded Software Systems. IEEE Transactions on Systems, Man, and Cybernetics. 1995;25: pp. 824-840.

[33] Guarro SB, Yau MK, Dixon S. Context-Based Software Risk Modeling: A Recommended Approach for Assessment of Software Related Risk in Nasa Missions. 11th International Probabilistic Safety Assessment and Management Conference and the Annual European Safety and Reliability Conference 2012, PSAM11 ESREL 2012. 2012. pp. 1839-1848.
[34] Guarro SB, Yau MK, Ozguner U, Aldemir T, Kurt A, Hejase M, et al. Formal Framework and Models for Validation and Verification of Software-Intensive Aerospace Systems. AIAA Information Systems-Infotech At Aerospace Conference, 2017, January 9, 2017 - January 13, 2017. Grapevine, TX, USA: American Institute of Aeronautics and Astronautics Inc, AIAA; 2017.
[35] Hegde J, Henriksen EH, Utne IB, Schjølberg I. Development of Safety Envelopes and Subsea Traffic Rules for Autonomous Remotely Operated Vehicles. Submitted to: Journal of Loss Prevention in the Process Industries. submitted: pp. 133-155.
[36] Hegde J. Tools and Methods to Manage Risk in Autonomous Subsea Inspection, Maintenance and Repair Operations. Trondheim, Norway: Norwegian University of Science and Technology (NTNU); 2018.
[37] Newman PM. Moos-Mission Orientated Operating Suite. Massachusetts: Department of Ocean Engineering, MIT; 2008. pp. pp. 53.

This page is intentionally left blank

Article 4

Thieme, C. A. & Utne, I. B. 2017. *A risk model for autonomous marine systems and operation focusing on human–autonomy collaboration. Proceedings of the Institution of Mechanical Engineers, Part O: Journal of Risk and Reliability,* 231, pp. 446-464, DOI: 10.1177/1748006x17709377

This page is intentionally left blank

Special Issue Article

Institution of MECHANICAL ENGINEERS

Journal of RISK AND RELIABILITY

Proc IMechE Part O:
J Risk and Reliability
2017, Vol. 231(4) 446–464
© IMechE 2017
Reprints and permissions:
sagepub.co.uk/journalsPermissions.nav
DOI: 10.1177/1748006X17709377
journals.sagepub.com/home/pio
SAGE

A risk model for autonomous marine systems and operation focusing on human–autonomy collaboration

Christoph Alexander Thieme and Ingrid Bouwer Utne

Abstract

Autonomous marine systems, such as autonomous ships and autonomous underwater vehicles, gain increased interest in industry and academia. Expected benefits of autonomous marine system in comparison to conventional marine systems are reduced cost, reduced risk to operators, and increased efficiency of such systems. Autonomous underwater vehicles are applied in scientific, commercial, and military applications for surveys and inspections of the sea floor, the water column, marine structures, and objects of interest. Autonomous underwater vehicles are costly vehicles and may carry expensive payloads. Hence, risk models are needed to assess the mission success before a mission and adapt the mission plan if necessary. The operators prepare and interact with autonomous underwater vehicles to carry out a mission successfully. Risk models need to reflect these interactions. This article presents a Bayesian belief network to assess the human–autonomy collaboration performance, as part of a risk model for autonomous underwater vehicle operation. Human–autonomy collaboration represents the joint performance of the human operators in conjunction with an autonomous system to achieve a mission aim. A case study shows that the human–autonomy collaboration can be improved in two ways: (1) through better training and inclusion of experienced operators and (2) through improved reliability of autonomous functions and situation awareness of vehicles. It is believed that the human–autonomy collaboration Bayesian belief network can improve autonomous underwater vehicle design and autonomous underwater vehicle operations by clarifying relationships between technical, human, and organizational factors and their influence on mission risk. The article focuses on autonomous underwater vehicle, but the results should be applicable to other types of autonomous marine systems.

Keywords

Risk modeling, autonomous underwater vehicles, human–autonomy collaboration, Bayesian belief network, autonomous marine system, human–autonomy interaction

Date received: 7 August 2016; accepted: 15 April 2017

Introduction

Autonomous marine systems (AMS), including autonomous ships, are the focus of ongoing industrial and academic research and innovation.[1–8] Recently, the Trondheimsfjord in Norway was opened as a test site for autonomous ships.[9] One requirement for AMS to operate in this area is that the risk has been assessed and it is demonstrated that the risk level is sufficiently low. Research projects, such as MUNIN[10] and AAWA[11] aim to establish concepts for autonomous cargo ships. Several small autonomous boats and vessels are already in use.[6,12–14] Autonomous underwater vehicles (AUVs) are an examples of AMS, which have been applied for more than two decades. They operate below the water surface and represent an important tool for scientific, commercial, and military purposes. They are able to map the sea floor, locate objects of interest, monitor and inspect undersea structures, and measure properties of the seawater.[15] Direct control below the water surface is difficult, due to the impediment of radio signals underwater and the low

Centre for Autonomous Marine Operations and Systems (AMOS), Norwegian University of Science and Technology (NTNU), Trondheim, Norway

Corresponding author:
Christoph Alexander Thieme, Centre for Autonomous Marine Operations and Systems (AMOS), Norwegian University of Science and Technology (NTNU), Otto Nielsens veg 10, NO-7491 Trondheim, Norway.
Email: Christoph.thieme@ntnu.no

communication bandwidth of underwater acoustics.[15] AUVs are able to adapt their mission paths to some extent to the environmental conditions to operate in the subsea environment and achieve the previously defined mission aim. Several shapes and types of AUVs exist. Yuh et al.[15] provide an overview of different AUVs and their purposes. In the future, AUV will be increasingly operated together with other autonomous systems, for example, autonomous aerial vehicles and surface vessels, for example, for joint monitoring of the environment.[16,17] In order to carry out such operations satisfactorily, AUVs need to be highly reliable. AUVs are expensive assets, often purpose built with a specific payload. A lost or misguided AUV might lead to failure of a mission, if no spare systems are available.[18] Therefore, risk models related to mission success (or correspondingly mission failure) are needed for decision support to the human operator.[19]

"Autonomous" does not mean that no personnel will operate them. Autonomy is a system's ability to change its pre-programmed plan of action to achieve its goal.[20] The degree of autonomy designed in a system is described by the level of autonomy (LOA). Several scales of LOA exist, see, for example Vagia et al.,[20] Insaurralde and Lane[21] and Wang and Liu.[22] Human operators monitor the AMS during a mission. They can change the mission plan, or abort a mission if necessary, for example, due to unforeseen changes in the operational conditions, or bad vehicle performance.[23] For example, the operators prepare the AUVs and make an overall mission plan, which might be erroneous.[24] Hence, informed risk models need to reflect these interactions. Utne and Schjølberg[25] identify relevant hazards related to human and organizational factors (HOFs) for AUV operation that should be considered in risk assessments. Ho et al.[26] discuss AUV operation and associated HOF that are relevant for a successful mission. Existing risk analyses of AMS mainly focus on the technical aspects and faults of AUV systems. Expert teams predict mission risk for the AUTOSUB AUVs based on the AUVs' fault logs.[27–30] A Markov model approach assesses the critical phases of operation.[24] Brito and Griffiths[31] present a Bayesian belief network (BBN) approach for AUV risk management. Griffiths et al.[32] apply an expert elicitation process to the fault logs of two REMUS 100 AUVs to predict mission risk for different scenarios.

A few publications focus on autonomous surface vessels. Rødseth and Tjora[33] present a risk-based design process for autonomous ships. Based on this approach, Rødseth and Burmeister[34] present a hazard analysis for autonomous ships through a scenario approach. They identify risk control options based on these scenarios. These risk control options aim at avoiding hazardous situations, but the interaction with the operators is not a concern. Kretschmann et al.[35,36] present the qualitative and the coarse quantitative risk assessment for the conceptualized ship of the MUNIN project. Regarding the qualitative risk assessment, they identify human error in remote operation and maintenance, foundering in heavy weather, and security issues as the main hazards. Some risk models for autonomous vessels address heavy weather conditions, such as Ono et al.[37] and Li et al.[38] Harris et al.[19] review models for risk assessment of AUV and similar systems. They assess the applicability of these models to multi-vehicle operations and conclude that a bottom-up approach to risk assessment is most suitable.

Only a few risk models, however, actually include HOF. Thieme et al.[39] present a risk management framework for AUV, including HOF in a coarse risk assessment of AUV. Thieme et al.[40] also present a qualitative BBN for AUV operation with focus on operator performance. None of the above-mentioned works, however, takes into account the important interaction between human operators and the technical system as a source for potential mission failure, which is addressed in this article.

Risk models considering HOF in AUV operation should treat the human operators and the autonomous system as collaborators, and not as individual or independent systems. Human–autonomy collaboration (HAC) can be defined as the cooperative and collaborative performance of the human operators and the autonomous system to achieve a goal jointly.[41] Hollnagel[42] argues that a model assessing human–machine systems requires a sound underlying model of the processes that happen during the interaction. This should reflect how the joint performance of human and machine is affected by the context and circumstances.[42]

The objective of this article is to present a BBN risk model focusing on HAC for AUV operation. The risk model should benefit users and manufacturers of AUVs and other AMS, to improve the design of these systems and support operator decisions during operation.[43] Since AMS may have similar requirements and demands as AUVs with respect to HAC, the risk model could be adapted to other AMS, as well. The BBN in this article extends the scope of Thieme et al.,[40] since quantification of the BBN and a case study are included. The case study gives insight into the usefulness and validity of the HAC BBN. The result of the research presented in the article shows that the two most efficient ways of improving HAC are through better training and inclusion of experienced operators, and through improved reliability of autonomous functions and situation awareness (SA) of vehicles. The HAC BBN is part of a larger future risk model for AUV operation, which considers environmental interactions, technical system performance, and regulatory and customer requirements, and enables assessment of mission success and the effect of risk control.

The next section describes the development process of the BBN. Then, the HAC BBN is presented, including a case study with quantification and validation. The discussion follows, before the last section concludes the article and states further work.

Development of the BBN

BBNs have been developed for risk assessments in various industries. In the marine domain, BBNs are applied for, for example, ship collisions,[44] ship groundings,[45,46] maintenance work on offshore installations,[47,48] and maritime transport systems.[49] BBNs are acyclic directed graphs and consists of nodes and arcs. Nodes have a set of variables, representing the state of the node. Arcs connect parent nodes with child nodes, representing the influence. Arcs are associated with conditional probability tables (CPTs) that determine the child nodes' states based on the parent nodes' states. Input nodes have no parent nodes, they are associated with a default probability to reflect their state. The Bayesian reasoning laws are used to update BBNs.[50] For more specific details on BBN, see, for example, Jensen and Nielsen[50] or Kjærulff and Madsen.[43]

The development of a BBN also includes some challenges. It is important to identify and include all relevant factors that influence risk in a BBN, as well as their relationship. A meaningful BBN model includes well-defined nodes, and the problem addressed in the model must lie within a structured domain with causal relationships.[43]

The development of the BBN in this article follows a five-step process:

1. Describe aim and context of the BBN;
2. Gather and group information relevant for the context into nodes;
3. Connect the nodes with directional arcs;
4. Determine the CPTs and quantify the model;
5. Test and validate the model.

Steps 1–3 are mainly based on the guidance on construction of BBNs by Jensen and Nielsen.[50] Steps 4 and 5 are adjusted to the purpose of the development of the HAC BBN. The BBN in this article was created with the computer program GeNIe 2.0 by the Decision systems laboratory, University of Pittsburgh, USA.[51] The following sub-sections explain the development process in detail.

Step 1: define aim and context of the risk model

The aim of the model in the article is to show the relationship between human operator performance and the technical performance of the autonomous system. The aim of the model determines the definition of the top node, which is HAC performance. HAC represents the joint performance of the human operator and the autonomous system during a mission of an AUV, its deployment or its retrieval. The presented model shall aid during the planning of an AUV mission to identify potential problems that might arise. The model in this article can also be used as an aid during the design of a system, since it highlights important relationships between the human operators and the technical system. The model shall be seen in the context of the operation of AUV described in section "Introduction."

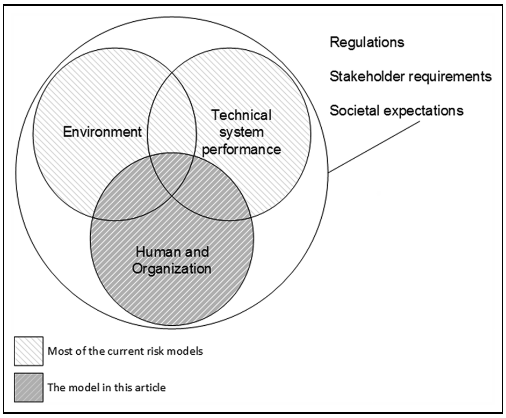


Figure 1. The main aspects to include in an overall risk model for AUV operation. The HAC model focuses on the human and organizational part.

Figure 1 shows that an overall risk model for AUV operation should include aspects related to the technical system, environmental conditions, and HOFs, that is, HAC. Regulations from the authorities, stakeholder requirements, and societal expectations are also issues that need to be considered. The HAC model is the scope of this article, since several works have already focused on the technical system performance and environmental conditions, as mentioned in the introduction. Future work remains to integrate all these aspects into one model.

Step 2: gather and group relevant information

Literature on human–autonomy interaction provides relevant information for the model in this article and determines the basis for the development of the nodes. Based on the definition of HAC, we may group the literature used to develop the model into two overall categories: (1) autonomy and automation and (2) HOFs in risk modeling. Table 1 summarizes the details of the literature and the references related to the nodes in the HAC BBN model. Qualitative influence models for use of automated functions were developed by Riley[78] and Parasuraman and Mouloua (cited in Parasuraman and Riley[56]). Donmez et al.[70] present a discrete simulation to determine operators' performance of supervisory control over multiple unmanned aerial vehicles and AUVs. These models are rather coarse and the former two do not contain recent findings. Therefore, it is necessary to aggregate recent findings in this domain and incorporate the considerations for AMS, that is, specifically for AUV operation in this article.

HOFs do not interact linearly.[62] Most methods used in probabilistic safety assessment are not suitable for assessing the HAC performance and a systemic

Table 1. Definition and description of the nodes included in the HAC BBN.

Node	Description	Reference in which this factor is mentioned
Autonomous Function Performance	Node summarizing the performance of autonomous functions of the system	N/A
Communication	Information exchange between operators to fulfill the assigned mission	52–54
Etiquette	"Set of prescribed and proscribed behaviours that permits meaning and intent to be ascribed to actions"[55] of the system	26, 55–57
False Alarm Rate	Rate of status messages that contain erroneous information	53, 56, 58, 59
Fatigue	"Inability [of the operator] to function at the desired level due to incomplete recovery from the demands of prior work and other waking activities."[60]	45, 56, 60, 61
Feedback from the System	Node summarizing the way a system gives feedback, to the operators, on status, intentions, and actions	52, 56–59, 62–65
Human–Autonomy Collaboration Performance	Node summarizing the overall performance of operators in conjunction with the autonomous functions of the system to achieve the mission goal	N/A
Human Operator Performance	Node summarizing the nodes that influence the human operators' performance	52, 54, 55, 57, 59, 62–64, 66, 67
Interface Design	Design principles applied to the physical and virtual interfaces of the system	53, 56, 57, 65, 68, 69
Level of Autonomy	The degree of the systems' ability to make independent decisions. This depends on the type of operation to be carried out and the type of AUV. This relationship is not further included in the model	26, 59, 61, 64, 67, 70, 71
Mission Duration	The duration of use and operation of AUVs for a mission. It also depends on the type of mission, type of vehicle, and the environmental condition. These interactions are not modeled, since they would require that environmental and technical aspects are fully included in the model	72
Number of Vehicles per Operator	Number of AUVs and AUV types, one operator operates concurrently	26, 54, 59, 61, 62, 64, 70, 71, 73
Operators' Experience	Level of experience of the operators with operation of the AUVs. This includes experience with AUV programming, AUV maintenance, AUV deployment and recovery, assessment of the marine environment, and working in the marine environment	55, 56, 63, 68, 74
Operators' Training	The amount of relevant training operators received for operation of AUVs. Relevant training includes training with respect to AUV programming, AUV deployment and recovery, AUV maintenance, the marine operation environment, and working in the marine environment	56, 58, 63, 68
Procedures	Provided documentation that prescribes operation and provides guidance to operator	68, 75
Reaction Time	Time, which the operators need to react to a situation that needs their attention	53, 58, 59, 71
Reliability of Autonomous Functions	The system's ability to perform its functions as required during the time of use. This includes mission relevant and diagnostic functions	53, 55–59, 64, 65
Shift Scheme	Pattern, which determines the operators' working and resting time	45, 60, 72
SA of Human Operators	Perception and comprehension of the AUVs' state and situation during operation by the operator, and projection of the future state	26, 53, 56, 59, 63, 67, 71, 76
SA of Vehicles	The vehicles' ability to perceive information, interpret, integrate, and assess relevance of that information, and predict the future with this information and prior background knowledge	77
Task Load	Number of tasks that have to be executed concurrently by one operator. This evaluation should include the consideration of complexity of tasks	52, 53, 56–58, 62, 68, 71
Time Delay of Transmission	Time that a message needs from the AUV to the operators or vice versa	26
Trust	"Users' willingness to believe information from a system or make use of it"[55]	26, 53, 55, 56, 57, 58, 64, 65, 74
Workload	The work demand encountered by the operators during AUV operation	26, 53–56, 58, 61–65, 67, 70, 71, 73, 74

SA: situation awareness.

approach is suggested.[42] BBNs are a useful tool for risk modeling, respecting the aforementioned considerations. They are traceable,[43] represent dependencies visually, can be used for prognosis and diagnosis.[44] Not only causal, but also uncertain dependencies in complex systems can be included.[79] Existing data and expert judgment can be combined and used to quantify BBN.[43,44] Furthermore, existing methods, such as fault trees and event trees, can be transformed into BBN, which means that modeling approaches can be combined.[44]

BBNs are also used for human reliability assessment (HRA), for example, see Mkrtchyan et al.[80] BBN versions of established methods, such as the SPAR-H method,[68,81] are more flexible and can be extended to model performance shaping factors (PSF) with more details, including task-specific knowledge. In HRA, the advantages of using BBN are causal and evidential reasoning, incorporation of information from different sources, graphical representation of causal relationships, and the possibility to include probabilistic modeling methods.[80] The existing literature gives confidence that BBN is a suitable tool to model risk of AUV operation, including HOF.

Step 3: connect the nodes

The arcs in the BBN model are developed based on the findings in the literature and the relationships identified between factors. These findings were merged, in order to determine the network. Some factors have a mutual influence on each other. This makes it difficult to define clearly these arcs. Since BBNs are acyclic, it is not possible to model mutual influences. In order to resolve mutual influences, the most frequently mentioned direction of influence define these otherwise ambiguous arcs.

Step 4: CPTs and case study

Several ways of CPT elicitation exist, for example, through theory, observed frequencies, or expert estimates.[50] A data-driven approach to deriving the CPTs is challenging for the model, since there is lack of data regarding HOF and AUV operation. Only a few investigation reports of loss of AUVs are available, for example, Strutt.[82] Direct elicitation of CPTs is resource intensive, but methods for reduced effort have been developed.[83] Vinnem et al.[47] use an approach based on building functions to assess CPTs. This process is modified and applied in this article because it reduces the amount of elicitation needed. The process focuses on assessing the strength of influence from parent nodes on their child nodes and on building templates. It is assumed that the parent nodes are independent. The adapted steps from Vinnem et al.'s[47] are as follows: (1) define templates for the CPT assessment based on triangular distributions, (2) determine the strength of influence of each parent node on the child node, and (3) combine the templates with the respective weights in the CPT of the parent node. For some nodes, the CPT assessment needs to be adapted for the HAC model; more details are given in section "Quantification of the BBN."

The data for the input nodes in the model in this article were derived in a case study, with basis in AUV operation in the Autonomous Underwater Robotics (AUR) Lab at the Norwegian University of Science and Technology (NTNU).

Step 5: validation

Validation provides assurance that the BBN reflects the system it shall represent and that outputs and mechanisms that produce these outputs reflect the real processes. Validation of BBN is challenging, simply applying a comparison to data or using experts to determine validity might overlook important aspects of model uncertainty.[84] Pitchforth and colleagues[84,85] propose a framework to validate BBNs structurally and quantitatively. This framework was chosen for this BBN, since data-driven validation is not possible. The suggested model in this article is compared to existing models, with respect to certain modeling aspects. The framework applies five tests in two categories: expert-based validation and data-based validation.

Expert-based validation consists of the following three tests:[84] (1) face validity assess the BBN's structure in comparison to what the literature or experts predict; (2) content validity tests, if all relevant factors are included in the model; and (3) convergent and discriminant validity assess if the model is similar to and different enough from other models with a similar aim for a different system. Data-based validation considers two aspects:[84] (1) concurrent validity, that is, the BBN's behavior in comparison to the behavior of (parts of) similar models and (2) predictive validity, that is, the BBN's estimations in comparison to available real-world data. As mentioned, no comprehensive data are available and therefore data-based validation is only limited possible. Details are stated in section "Validation of the model."

The HAC risk model

The BBN and description of the nodes

HAC depends on the autonomous functionality designed into the technical system, the human operators, the interaction between the technical system and the human operator, and the organization in which the operators act.[41] An adequate HAC is associated with a high probability for a successful mission. Figure 2 shows the HAC BBN. Table 1 describes the nodes in the BBN, including references to the associated literature. The next paragraphs describe the network in more detail. The literature provides the basis for the arcs and the relations between the nodes.

Human operator performance in cooperation with an autonomous system is widely researched. It is

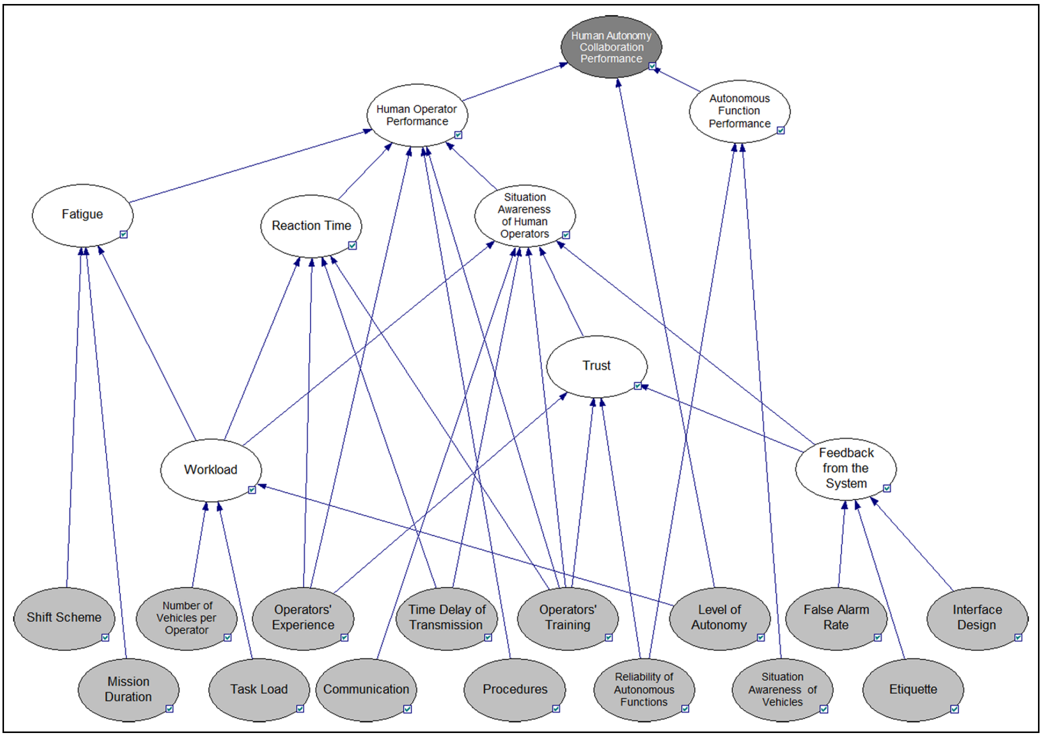

Human Autonomy Collaboration Performance
Human Operator Performance
Autonomous Function Performance
Fatigue
Reaction Time
Situation Awareness of Human Operators
Trust
Workload
Feedback from the System
Shift Scheme
Number of Vehicles per Operator
Operators' Experience
Time Delay of Transmission
Operators' Training
Level of Autonomy
False Alarm Rate
Interface Design
Mission Duration
Task Load
Communication
Procedures
Reliability of Autonomous Functions
Situation Awareness of Vehicles
Etiquette

Figure 2. BBN for human–autonomy collaboration performance.
Node color-coding: light grey, input nodes; white, intermediate nodes; dark grey, HAC node.

influenced by Trust, Reaction Time of the Operators, Procedures, Fatigue, SA of Human Operators, Workload, Operators' Training, and Operators' Experience.[26,52,55–59,61–64,66,67,71,72,74,75,78] Experience and training refer to all operational aspects of AUV operation. This includes AUV programming, AUV maintenance, AUV deployment and recovery, assessment of the marine environment, and working in the marine environment.

Research of HAC focuses on SA. Low SA of Human Operators is a symptom of low levels of other HOFs.[63] SA of Human Operators is influenced by Trust, Workload, Feedback from the System, Time Delay of Transmission, Communication, and Operators' Training.[26,52–54,56,63,67]

Trust in the system is built with time through the Operators' Experience with the system.[65] Trust also depends on the operators' Workload, Feedback of the System, and Reliability of Autonomous Functions.[26,53,55–57,64,65] Workload and Time Delay of Transmission influence the Reaction Time of operators.[26,53,58,59,71] Operators' Experience and Training determine familiarity with the systems and influence the Reaction Time. The Operators' Workload depends on the amount and kind of tasks they have to carry out.[70] In the model, Workload is determined through the LOA, Task Load, and Number of Vehicles per Operator.[26,52–54,56,59,62,64,66,70,71]

Gander et al.[60] highlight the necessity to consider fatigue in risk management. Akhtar and Utne[45] analyze the influence of fatigue on risk in maritime transport. Fatigue depends on the Workload, Mission Duration, and the Shift Scheme.[45]

Feedback of the System summarizes the system's way of presenting information to the operators, through Etiquette, False Alarm Rate, and Interface Design,[26,53,56,57,65] SA of Vehicles and Reliability of Autonomous Functions constitute the Autonomous Function Performance. Autonomous functions are those functions that the AUV carries out to finish a mission successfully. This includes mission relevant functions, for example, sensing of the environment, data recording, and diagnostic functions, which are necessary for the AUV to follow and adapt its mission plan to achieve the most satisfactory mission outcome. SA of Vehicles influences the Autonomous Function Performance, since it is the AUVs' ability to perceive and analyze their own situation and predict their future situation.[77] A low Reliability of Autonomous Functions implies that the system does not execute its functions when needed and in the right way.

States of the nodes

Table 2 presents the proposed states for the nodes described in Table 1. Proposals of evaluation criteria

Table 2. Proposed states for the nodes in the human–autonomy collaboration performance BBN.

Node	Proposed states
Autonomous Function Performance	Low, medium, high
Communication	Low, adequate, high (e.g. no communication of relevant information; communication of relevant information; clear and unambiguous communication of all relevant information)
Etiquette	Disruptive, mediocre, good (e.g. intrusive messages with abstract information; messages partly fulfil design criteria from Sheridan and Parasuraman,[57] p. 102; messages fulfil design criteria from Sheridan and Parasuraman,[57] p. 102)
False Alarm Rate	High, medium, low (e.g. more than one of 1000 status updates is erroneous; one status update of between 1000 and 10,000 is erroneous; less than one of 10,000 status updates is erroneous)
Fatigue	High, medium, low
Feedback from the System	Poor, mediocre, good
Human–Autonomy Collaboration Performance	Inadequate, adequate
Human Operator Performance	Low, medium, high
Interface Design	Poor, mediocre, good (e.g. no interface design principles applied; ecological interface design principles partly applied; ecological interface design principles fully applied.[57])
Level of Autonomy	LOA 1, manual control; LOA 2, action support; LOA 3, batch processing; LOA 4, shared control; LOA 5, decision support; LOA 6, blended decision-making; LOA 7, rigid system; LOA 8, automated decision-making; LOA 9, supervisory control; LOA 10, full autonomy (based on Endsley and Kaber[66])
Mission Duration	Long, medium, short (e.g. more than 8 h; between 4 and 8 h; less than 4 h)
Number of Vehicles per Operator	High, medium, low (e.g. more than three vehicles or vehicle types; between two and three vehicles or two vehicle types; less than two vehicles)
Operators' Experience	Low, medium, high (e.g. less than half a year, between half a year and 1 year; more than 1 year)
Operator' Training	Low, adequate, high (e.g. operators have not attended required trainings; operators have gone through required training; additional to required trainings, additional training was attended)
Procedures	Poor, adequate, good (e.g. procedures are incomplete; procedures are covering all expectable situations; procedures are well written covering all expectable situations and give guidance in case of unforeseen events)
Reaction Time	Long, medium, short
Reliability of Autonomous Functions	Low; mediocre, high (e.g. $\leqslant$95%, $>$95% and $\leqslant$99%, $>$99%)
Shift Scheme	Variable working hours; 8–4–4–8; 12–12 or 6–6 (hours on and off duty, based on Akhtar and Utne[45])
SA of Human Operators	Low, medium, high
SA of Vehicles	Low, medium, high (e.g. basic perception of the environment; interpretation, integration and ranking of perceived information; prediction of future situations, with available knowledge and perceptions, based on Endsley[76])
Task Load	High, medium, low (e.g. more than three nominal tasks, or more than one moderately complex tasks, or one or more highly complex tasks; between two and three nominal tasks, or one moderately complex task; two or less nominal tasks)
Time Delay of Transmission	Long, medium, short (e.g. more than 40 s, between 40 and 20 s, shorter than 20 s)
Trust	Distrust, adequate, overreliance
Workload	High, medium, low

are given for the input nodes. The states are arranged from "worst" to "best" states, except for LOA, and Trust. States that need clarification are described below.

The HAC node has the states "Inadequate" and "Adequate." This represents the combined expected performance of the operators and the AUV system. An "Adequate" HAC can be expected to contribute to a higher probability of mission success. An "Inadequate" HAC is associated with a lower expected performance, for example, errors by the operators or inadequate decisions by the autonomous system. It has a negative influence on mission success, and the probability for negative mission outcomes increases, for example, loss of an AUV.

The "Low" states of Reliability of Autonomy Functions is based on the assumption that a reliability below 95% is not acceptable and performance decreases strongly below 95%.[64] No manual control or correction is possible. Therefore, this threshold was selected. The states "Medium" and "High" are exemplarily given.

The states of Shift Scheme in Table 2 need explanation: Akhtar and Utne[45] show that in the presence of other fatigue-related factors, the "8–4–4–8" scheme contributes more to fatigue than the shift schemes "12–12 or 6–6." Variable working hours, however, may lead to more fatigue.

Quantification of the BBN

The process for CPT assessment was adapted from Vinnem et al.[47] The first step is to define the templates used for CPT elicitation, which are based on a

Table 3. Discretized CPT templates for low and high strength of influence. Worst, intermediate, and best represent the states generically.

Parent's state	Child's states	Low strength template	High strength template
Worst	Worst	0.60	0.90
	Intermediate	0.30	0.09
	Best	0.10	0.01
Intermediate	Worst	0.20	0.05
	Intermediate	0.60	0.90
	Best	0.20	0.05
Best	Worst	0.10	0.01
	Intermediate	0.30	0.09
	Best	0.60	0.90

triangular distribution. Table 3 shows the CPT templates for assessment of the child nodes. The strength of influence defines the spread in the template for a given parent state. In this article, two strengths (low and high) are used. The templates are based on discretized triangular functions, which is a simplification from the original process in Vinnem et al.,[47] due to limited data available. A high influence template has a lower spread over the range of states. The range of states is referred to as Worst, Intermediate, and Best. These states correspond to the states presented in Table 2.

In the second step, the strength of influence of each parent node is assessed for the child node. For example, the Autonomous Function Performance has the parents Reliability of Autonomous Functions and SA of Vehicles, with corresponding states in Table 2. The strength of influence from Reliability of Autonomous Functions is rated high, since AUVs are highly dependent on the correct performance of their functions to execute a mission. SA of Vehicles is also rated as highly influential, since the operational picture is highly relevant for the AUVs to carry out their assigned functions appropriately.

The strength of influence also determines the weight of each parent node. A low strength of influence is associated with a weight of 1. A high strength of influence is associated with a weight of 3. The weights for each parent node are normalized with the total sum of all weights. The templates for each parent node are multiplied with their normalized weights to build a child node's CPT. For a given combination of the parent nodes' states, the weighted templates are added together and inserted in the respective column of the child node's CPT. This represents the third step of Vinnem et al.'s approach. In the above example, the high strength templates in Table 3 are used.

As an example of the elicitation process, consider the node Autonomous Function Performance. The strength of influence is considered the same for both parent nodes, that is, Reliability of Autonomous Functions and SA of Vehicles, and therefore, they are equally weighted. Table 4 shows the resulting CPT for the node Autonomous Function Performance. A small example demonstrates the calculation, the combination of states was chosen in order to clearly distinguish the contribution from the parents. For example, the CPT entry for "Low" Autonomous Function Performance for the combination of "Mediocre" Reliability of Autonomous Functions and "Low" SA of Vehicles is 0.475. Both, Reliability of Autonomous Functions and SA of Vehicles have a high influence on Autonomous Function Performance. Therefore, they are associated with a weight of "3" and the high strength templates in Table 3. The entry in the CPT is the sum of the contribution from the "Low" Autonomous Function Performance multiplied with the normalized weight $(0.05 \cdot (3/(3+3)) = 0.025)$ and the contribution from "Mediocre" Reliability of Autonomous Functions multiplied with the normalized weight $(0.9 \cdot (3/(3+3)) = 0.45)$. This process is repeated for all possible combinations of the two parent nodes' states for each state of Autonomous Function Performance. Appendix 1 contains the other strength of influence assessments of the parent nodes on the child nodes.

A few CPTs need a separate process, that is, the HAC node, Trust, and Workload. The CPT for the HAC node needs a separate process, as the templates cannot be applied and the LOA needs to be considered separately. Table 5 shows the CPT template used for the HAC node, since the templates from Table 3 are not suitable for translating directly the states "Low," "Medium," and "High" to "Inadequate" and "Adequate." In Table 5, "Low" performance of the Human Operator and the Autonomous System are mainly associated with an "Inadequate" HAC. Similarly, a "Medium" performance is mainly associated with an "Adequate" HAC. A "High" performance is strongly associated with an "Adequate" state.

Table 4. CPT of autonomous function performance.

Reliability of autonomous functions		L			Mediocre			H		
SA of Vehicles		L	M	H	L	M	H	L	M	H
State of Autonomous Function Performance	L	0.900	0.475	0.455	0.475	0.050	0.030	0.455	0.030	0.010
	M	0.090	0.495	0.090	0.495	0.900	0.495	0.090	0.495	0.090
	H	0.010	0.030	0.455	0.030	0.050	0.475	0.455	0.475	0.900

L: low; M: medium; H: high.

Table 5. CPT template for determination of the CPT of the human–autonomy collaboration performance node.

HAC state	State of Autonomous Function Performance or Human Operator Performance		
	Low	Medium	High
Inadequate	0.90	0.10	0.01
Adequate	0.10	0.90	0.99

Table 6. Proposed weights for building the CPT for autonomy collaboration performance depending on LOA.

LOA	Weight for	
	Autonomous Function Performance	Human Operator Performance
1	0.05	0.95
2	0.15	0.85
3	0.25	0.75
4	0.35	0.65
5	0.45	0.55
6	0.55	0.45
7	0.65	0.35
8	0.75	0.25
9	0.85	0.15
10	0.95	0.05

The LOA, by definition, proportions the influence from the human operator and the autonomous system on decision-making and performance. Hence, LOA determines the weight of the Human Operator Performance in relation to Autonomous Function Performance. Table 6 shows the LOA-dependent weights. They are based on the assumption that the human operators have most influence on the state of HAC when the AUV has a low LOA. Their influence decreases with increasing LOA. However, the Autonomous Function Performance is neither negligible at LOA 1, nor the Human Operator Performance at LOA 10.

The building of the CPT for Trust needs considerations, due to its three states. The literature[55–57,65] shows how "Distrust," "Overreliance," and "Adequate" Trust are formed. The states of Reliability of Autonomous Functions ("Low," "Mediocre," and "High") are directly associated with the respective formation of "Distrust," "Adequate" Trust, and "Overreliance." "Poor" Feedback from the system leads to "Distrust." A "Good" Feedback will lead to an "Adequate" level of Trust. Consequently, "Mediocre" feedback will lead to "Overreliance," since the operator might overlook cues. "Low" Operators' Experience leads to "Distrust." "High" Operators' Experience creates an "Adequate" level of Trust. "Medium" Operators' Experience is associated with "Overreliance." Similarly, "High" Operators' Training creates "Adequate Trust." "Low" operators training leads to "Distrust." "Adequate" training is associated with "Overreliance," since not all situations that would require the operators' attention are trained. This means that Trust has two states that have a negative influence on the operator.[55–57] These are "Distrust" and "Overreliance." Hence, the template for the "worst" state is used for both "Distrust" and "Overreliance" to build the CPT for SA of human operators.

The CPT for Workload needs additional assumptions due to its parent LOA. A lower LOA implies more work for the human operators. Hence, "LOA 1" to "LOA 3" were associated with a "High" Workload. "LOA 4" to "LOA 7" imply cooperation in execution of the operation and a "Medium" Workload. "LOA 8" to "LOA 10" represent the best possible state, and imply a "Low" Workload, since autonomous functions carry out most of the work.

Case study

NTNU operates one REMUS 100 AUV, designed and produced by Hydroid, through its Advanced Underwater Robotics Laboratory (AUR Lab).[86] The AUV is used for testing scientific equipment, surveys of the seabed, and biological and physical studies of the fjords of Norway. The data in the case study are mainly derived from earlier works.[39,87] and supplemented with information from the AUR Lab, the supplier,[88] and other publications.[32,89,90] The case study focuses on the operation phase of the mission to have sufficient data. Deployment and retrieval can be assessed by changing the states of the input nodes, according to the operators and mission states. However, insufficient information is available for these phases and a quantification in the case study is impossible.

Table 7 summarizes the states for the input nodes and related references used in the case study. LOA, Shift Scheme, and Number of Vehicles are deterministic, their state is known, and hence the probability is set to 1. Thieme[87] presents the rating of PSF for the SPAR-H method by two operators of the AUR Lab. Six undesired events are related to operators interacting with the REMUS 100 AUV. These events are as follows: AUV is not properly monitored, unexpected behavior is not detected, existing faults are not completely solved before deployment, faults are not recognized during planning phase or before deployment, wrong use of software leads to wrongly implemented parameters, and implementation of mission path or map is done wrongly. For a detailed description, see Thieme.[87] The PSFs of these events were assessed to be in either a low or poor state, an adequate or nominal state, or a good or helpful state. It was assumed that these ratings of the PSF correlate to the generic states in this article: Worst, Intermediate, and Best, respectively. The number of ratings was normalized over these states. The PSF ratings were used for the nodes Communication, Etiquette, Interface Design, Operators'

Table 7. States of the input nodes for the case study. For states without available reference (NA: not available), assumptions had to be made based on experiences in the AUR Lab.

Node	States			Comment	References
	Worst	Intermediate	Best		
Communication	0.001	0.749	0.250	Based on the PSF ratings of work processes	Thieme[87]
Etiquette	0.167	0.750	0.083	Based on the PSF ratings of Ergonomics/ HMI	Thieme[87]
False Alarm Rate	0.200	0.600	0.200	No data are available. A Medium False Alarm Rate is assumed, with low confidence	NA
Interface Design	0.167	0.750	0.083	Based on the PSF ratings of Ergonomics/ HMI	Thieme[87]
Level of Autonomy	LOA 7			AUV are pre-programmed, the software for programming assists in planning and mission implementation. This corresponds to LOA7	NA
Mission Duration	0.050	0.900	0.050	Missions were in average between 4 and 5 h (assuming a speed of 1.5 m/s and length of 25 km)	Thieme and colleagues[39,87]
Number of Vehicles per Operator	0.000	0.000	1.000	The AUR Lab operates one REMUS 100 AUV	Thieme and colleagues [39,87]
Operators' Experience	0.667	0.250	0.083	Based on the PSF ratings of Experience/ Training	Thieme[87]
Operators' Training	0.667	0.250	0.083	Based on the PSF ratings of Experience/ Training	Thieme[87]
Procedures	0.001	0.166	0.833	Based on the PSF ratings of Procedures	Thieme[87]
Reliability of Autonomous Functions	0.200	0.600	0.200	Griffiths et al.[32] report that 14.8% of mission were aborted preliminary by the REMUS 100. The exact reasons are not stated. Therefore, it is assumed that Reliability of Autonomous Functions is mainly Mediocre, with low certainty	Griffiths et al.[32]
Shift Scheme	0.000	0.000	1.000	Normally operators work a 12–12 shift scheme	NA
SA of Vehicles	0.050	0.900	0.050	The AUV is equipped with various sensors. Based on measurements, it assesses its own situation with simple reasoning. Therefore, it is assumed medium with high certainty	Hydroid[88] and Hagen et al.[89]
Task Load	0.001	0.916	0.083	Based on the PSF ratings of Complexity	Thieme[87]
Time Delay of Transmission	0.010	0.090	0.900	Messages can be delayed by more than 10 s. It was assumed that only a low percentage is delayed by more than 20 s	Ho et al.[90]

Experience, Operators' Training, Procedures, and Task Load.

For states of the nodes that have zero probability, since the operators in Thieme[87] did not use corresponding PSF ratings, a small probability was inserted in the current case study to reflect uncertainty. For the other states, available information from other works[32,39,87–90] was used to assess the most likely state. For some nodes, no references were available (marked with NA). These nodes are False Alarm Rate, LOA, and Shift Scheme. For these states, assumptions were made based on the experience with the AUR Lab. Based on the strength of knowledge, the strength of influence templates from Table 3 were used to derive the input probabilities.

Using the probabilities from Table 7 for the input nodes and updating the network in GeNIe, gives a probability of 28.5% for an "Inadequate" HAC state, and a probability of 71.4% for an "Adequate" HAC state. The probability of mission success decreases with an increased probability of "Inadequate" HAC (cf. Figure 1). Hence, the results of the case study imply that there is room for improvement. The HAC should be as "Adequate" as possible. A sensitivity analysis in the next section gives input to how the state of HAC could be improved.

Sensitivity analysis

GeNIe 2.0 was used to conduct a sensitivity analysis. The built in sensitivity analysis function of GeNIe 2.0 varies each node over the whole range and assesses the impact of this change on the target node. The target node for the sensitivity analysis is in this case the

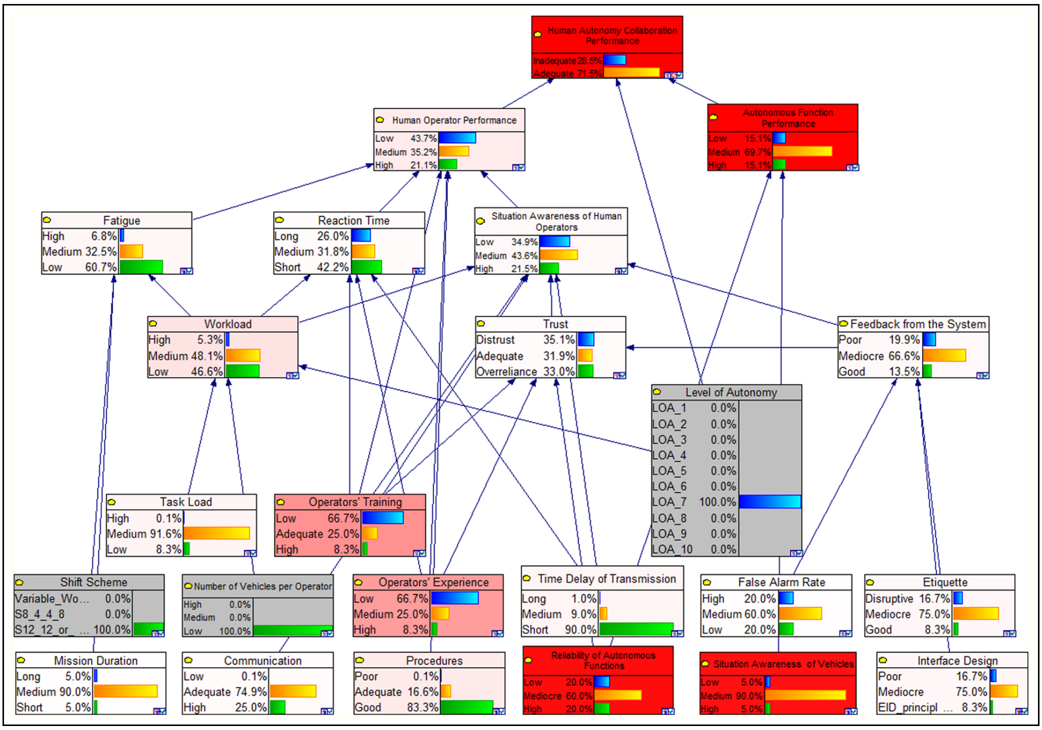


Figure 3. Sensitivity of the HAC node to input from its parent nodes. Dark red areas indicate a higher influence. Grey nodes are deterministic. The sensitivity from these nodes was not assessed.

Human–Autonomy Performance Collaboration node. Figure 3 shows the analysis results. Intensive red areas indicate a higher influence of nodes. The most influential input nodes on the HAC node are Autonomous Function Performance, Reliability of Autonomous Functions, SA of Vehicles, Operators' Training, and Operators' Experience. The nodes LOA, Shift Scheme, and Number of Vehicles per Operator are deterministic and depend on the mission. Hence, their influence could not be assessed during the sensitivity analysis. Figure 4 shows the effect of changing the states of each node in the case study on the probability of "Adequate" HAC. The case study is shown as reference value, as well as the Best Case and the Worst Case. For the Best Case and Worst Case, all input nodes that were not deterministic were set to their best and worst states, respectively. If all input nodes are in their best state, the probability of an "Adequate" HAC is 95.1%. With the input nodes in their worst states, the probability of "Adequate" HAC drops to 23.4%. The CPT of HAC limits the best and worst probability of HAC. This is discussed in the section "Discussion."

To assess the influence of the individual nodes, they were set individually to the best and worst cases. Figure 4 is arranged such that the most influential nodes are on the top and the least influential on the bottom. Figures 3 and 4 show that Reliability of Autonomous Functions and SA of the Vehicles are the most influential nodes in the case study. In their worst state, they reduce the probability of an "Adequate" HAC by more than 25%.

The best state of Reliability of Autonomous Function and SA of the Vehicles improves the probability of "Adequate" HAC by 7.1% and 4.4%, respectively. Operators' Training and Operators' Experience are the most influential human factors in the case study. Their worst states reduce the probability of "Adequate" HAC by 2.5% and 2.2%, respectively. The best states improve the probability of "Adequate" HAC by 5.8% and 5.3%, respectively. The states with the least influence are Communication, Mission Duration, and False Alarm Rate. Their best states do not improve the probability of "Adequate" HAC. However, the worst states decrease the probability of "Adequate" HAC by 0.2%, 0.1%, and 0.1%, respectively.

Validation of the model

Six publications form the basis of the validation, that is,[31,46,47,68,70,78] These publications cover similar models and considerations as the model in this article. It is assumed that face validity is established by the iterative building of the BBN from the literature, that is, structurally, the model is similar to Riley.[78]

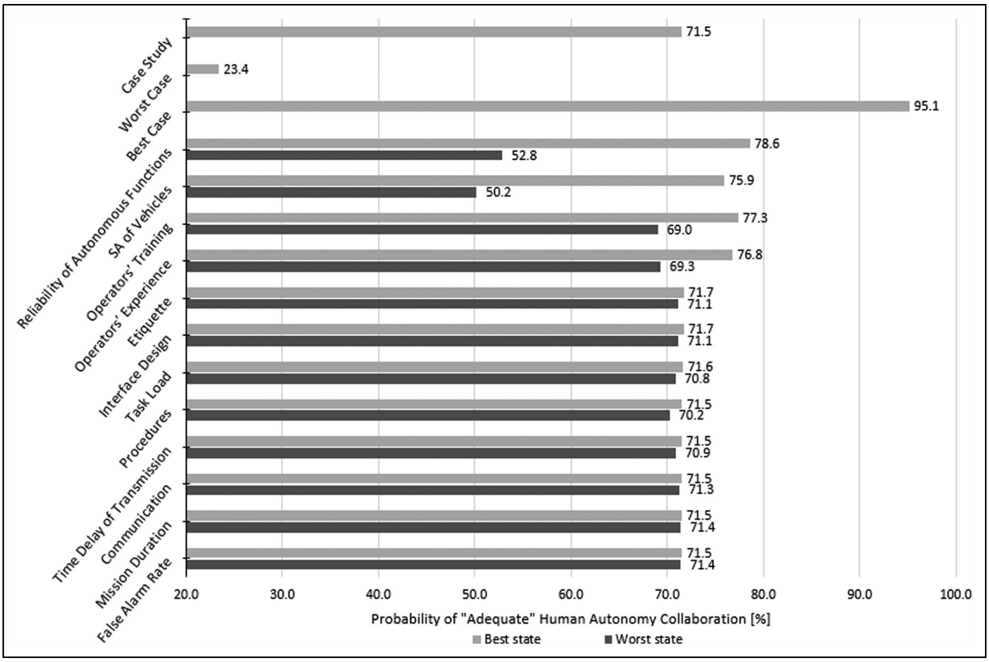


Figure 4. Effect of changing the states of the nodes individually on the probability of "Adequate" human–autonomy collaboration performance. The Worst Case and the Best Case refer to the nodes being set in the worst and best state combined.

Each node in the model presented in this article, except LOA and HAC, has three states. Brito and Griffiths[31] use more states for their nodes, which reflect discretized physical conditions and risk classes. They do not include nodes, which reflect HOFs. This makes a comparison difficult. Groth and Swiler[68] use three and five states. Mazaheri et al.[46] use nodes with mainly two states and few with three states. Content validity is assumed, since the relevant literature, which includes HOF,[46,68] uses similar states and discretization as in the BBN presented in this article.

The CPT assessment process was modified from Vinnem et al.,[47] with simplified weights and CPT templates. The parametrization process seems valid, since it was adopted from the literature and leads to the expected model behavior. The presented model is a sub-model to find the mission success of AUV operation and it models considerations that are not included in Brito and Griffiths.[31] Hence, there is no convergence. Since this article focuses on AUV operation, it can be compared to the model of Mazaheri et al.[46] with respect to discriminant validity. Their article focuses on ship groundings and includes specific nodes, which are not present in the HAC BBN. Discriminant validity is assumed.

Donmez et al.[70] present results for the performance of operators operating different types of autonomous vehicles. A comparison is not possible, since the case study is based on operation of one AUV and the presented model in this article does not assess HAC as a percentage of Score, as Donmez et al.[70] Concurrent validity cannot be established, since there are no suitable reference models.

The model produces expected outputs regarding the overall model behavior in the case study. Setting the input nodes to their best states resulted in a high probability of "Adequate" HAC of 95.1%. Setting the variable input nodes to the worst case in the case study results in 23.4% probability of "Adequate" HAC. The presented HAC BBN model is sensitive to the input (section "Sensitivity analysis"). The model reflects, for example, that the Reliability of Autonomous Functions and the Operators' Experience and Training are very influential, as was found in the literature.[56,57,91] AUV have a high LOA, this is reflected by the fact that the Reliability of Autonomous Functions and SA of the Vehicles modify the probability of "Adequate" HAC most strongly. In addition, HOFs, such as, mission duration, communication, and procedures, influence the probability of "Adequate" HAC only marginal. This is an expected behavior of the model for a high LOA. This gives confidence that the model reflects the real world.

Thieme and Utne[92] analyze, among others, mission and fault logs of nine mission of the REMUS 100 of the AUR Lab. One of these missions had to be aborted due to thruster failure. Unfortunately, no documentation or investigation of the aborted mission and its circumstances exist, which means that it is difficult to use for validation. Incidents and operations need to be better documented in order to derive a sound basis for network validation. Data are missing to establish predictive validity with respect to numerical verification of the outputs.

Discussion

The HAC BBN in this article is developed specifically for AUV operation and merges the findings from the human–autonomy interaction literature. The case study shows that the HAC BBN is able to produce meaningful results. The sensitivity analysis shows that HAC in the case study can be improved most significant in two ways: (1) through better training and inclusion of experienced operators and (2) through improved Reliability of Autonomous Functions and SA of Vehicles. However, the HAC BBN is only a sub-model of the overall risk model (Figure 1) and its influence on mission success remains to be modeled.

Although the model is sensitive to changes in most of the input nodes, some of them only have a minor influence on the state of HAC. These input nodes are Communication, Etiquette, False Alarm Rate, Interface Design, Mission Duration, Task Load, and Time Delay of Transmission. These nodes are associated with Human Operator Performance. Their low influence can be attributed to the LOA of the AUV, which is high and limits the influence of Human Operator Performance on the HAC node.

Regarding the case study, the input data were adapted from the literature and complemented with information gathered from the AUR Lab. Especially, Operators' Experience and Training are rated low. The data used were gathered after only 12 missions in the Lab. A separate assessment from the data used for training and experience was not possible. Hence, data from more recent operations may give a better estimate of the state of HAC. The presented results need to be considered with care.

The CPT templates were derived based on approximated and discretized triangular distributions. This is a simplification from the original method, in Vinnem et al.[47] This adaptation was necessary, since the original method uses six states. This article only uses three states, due to the lack of data. The influence of the strength the template on the result could not be assessed. More investigation is necessary in order to verify the applicability of the chosen weights and templates. One node for which a refined elicitation process is necessary is Trust, due to the opposing states Distrust and Overreliance. In this case, specially adapted templates might overcome this issue. The weighing between Human Operator Performance and Autonomous Function Performance is assumed linearly dependent on the LOA. Research focuses only on few LOA. No comprehensive data are available to derive these weights. Simulator studies similar to Donmez et al.'s[70] should be carried out in order to validate the quantification of the model and gain an improved model parametrization.

Fatigue-related considerations are transferred from Akhtar and Utne,[45] who investigate crews of cargo vessels. However, this article adapts their findings. More investigation is necessary in order to validate the applicability of their findings.

Workload is a complex research topic. Each operator will perceive Workload differently.[93] Hence, the Workload node in the HAC BBN depends only on the tasks to be executed. Workload influences Trust, a higher Workload creates "Overreliance."[56,65] Contrary, if an operator shows "Distrust" toward the autonomous system, the workload is increased due to more frequent and detailed checks.[26] This shows that there is a mutual influence, which is not possible to model with BBN.

Some HOFs mentioned in the literature were excluded, since they were considered not applicable: the operators' fitness for duty and individual personalities[68,60,94] are only partially included, for example, through Fatigue, since little research on this topic in relation to human automation interaction and AUV is available. The operators' confidence in their own abilities in relation to the autonomous capabilities[55–57,65,91] are not included explicitly, this is assumed part of Operators' Experience as an adequate confidence develops with experience. The operators' perceived risk associated with the task to execute[55–57,65] is excluded, since it is associated with high-risk industries, such as nuclear power plant operation or aviation. It is also connected with the possibility of not using automated functions, which is not possible for AUVs.

Direct influences from the environment have been neglected in the model. Nevertheless, these will inevitably influence the operator if they operate the AUV from a ship. If AUV operation is shore based, the direct influence of weather and sea state is minor to the operator, but may impact the technical system (AUV). The HAC BBN does not address these issues. First, the examined literature does not cover these relations completely. Second, the environment, that is, weather and sea state, affects not only the operators and the autonomous function performance, but also the technical performance, and technical factors influencing HAC. Assessment of these factors and interactions requires a holistic system view. This would overextend the scope of this article.

Conclusion and further work

This article presents a detailed BBN for HAC performance for AMS. The case study and development focus on AUV operation. The BBN can be used for assessment of mission success of AMS operation, during the planning and preparation phases. The relevant nodes were identified in the literature and their relationships modeled, accordingly. A case study on AUV operation, based on information from NTNU's AUR Lab, was used to assess the BBN's applicability. It shows that the HAC BBN is sensitive to input and produces reasonable results. Validity is assumed for the structure, discretization, and parametrization. Data-based validation is difficult to establish due to limited data, but is assumed, since the models behave as expected.

The case study shows that the probability of an "Inadequate" HAC is 28.5% and consequently, 71.5% for an "Adequate" HAC. A sensitivity analysis shows that SA of the autonomous vehicles and the reliability of autonomous functions are among the most influential input nodes, which gives confidence that the model reflects the real world. This has implications for the design of autonomous vehicles, which need to ensure efficient cooperation between the operators and potentially other autonomous vehicles. A reliable and self-aware system will promote improved mission performance. In addition, the sensitivity analysis shows that Operators' Experience and Training are highly influential on the state of HAC. The human operator cannot be neglected and is a decisive factor in AUV operation.

Nodes included in this model, which were not mentioned previously in the literature in connection with operation of AUV and human–autonomy interaction, are Human Fatigue, Shift Scheme, and SA of Vehicles. The BBN was developed based on an extensive literature study. Work similar to Donmez et al.,[70] which assess the influence of certain factors on the mission outcome, can aid in validating and improving the model. AUV simulators are a useful tool for these kind of assessments, which should be carried out in the future. In addition, investigation of incidents and their documentation can help in this validation process.

The BBN is adaptable to other AMS, such as underwater gliders or autonomous surface vehicles. The tasks and modes associated with operation of these type of AMS are similar to the operation of AUV. They are remotely supervised and intervention is necessary only in few cases. Some of the nodes' states might need adaption to the specific cases of these other systems. Necessary adaptations to other systems need to be further investigated in the future.

The HAC BBN presented in this article could be part of a larger overall risk model for the assessment of the probability of mission success. Further work is necessary to integrate it completely with the other model considerations: environmental interactions, technical system performance, societal expectations, and regulatory and customer requirements. The BBN modeling technique and the chosen quantification method are useful tools for implementation of these aspects.

Acknowledgements

The authors would like to thank the two anonymous reviewers who contributed with valuable comments, suggestions, and questions on the first version of this article. The authors wish to acknowledge Professor Ingrid Schjølberg for her comments on an earlier version of this article.

Declaration of Conflicting Interests

The author(s) declared no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

Funding

The author(s) disclosed receipt of the following financial support for the research, authorship, and/or publication of this article: This work was supported by the Research Council of Norway through the Centres of Excellence funding scheme, Project number 223254—AMOS.

References

1. United States Department of the Navy. *The Navy Unmanned Undersea Vehicle (UUV) master plan*. Washington, DC: United States Department of the Navy, 2004.
2. United States Department of the Navy. *The Navy Unmanned Surface Vehicle (USV) master plan*. 1st ed. Washington, DC: United States Department of the Navy, 2007.
3. Huntsberger T and Woodward G. Intelligent autonomy for unmanned surface and underwater vehicles. In: *Proceedings of the OCEANS 2011*, Kona, HI, 19–22 September 2011. New York: IEEE.
4. Martins R, De Sousa JB, Carvalho Afonso CC, et al. REP10 AUV: shallow water operations with heterogeneous autonomous vehicles. In: *Proceedings of the OCEANS 2011 IEEE—Spain*, Santander, 6–9 June 2011. New York: IEEE Computer Society.
5. Faria M, Pinto J, Py F, et al. Coordinating UAVs and AUVs for oceanographic field experiments: challenges and lessons learned. In: *Proceedings of the IEEE international conference on robotics and automation*, Hong Kong, China, 31 May–7 June 2014, pp.6606–6611. New York: IEEE.
6. Bertram V. Autonomous ship technology -smart for sure, unmanned maybe. In: Morgan G (ed.) *Smart ship technology*. London: The Royal Institute of Naval Architects, 2016, pp.5–112.
7. Flæten SØ. Dette skipet er utslippsfritt og har ingen mennesker ombord [This ship is emission free and has no people on board]. *Teknisk Ukeblad*. Teknisk Ukeblad Media AS, 20 September 2014, https://www.tu.no/artikler/dette-skipet-er-utslippsfritt-og-har-ingen-mennesker-ombord/231695 (accessed 23 July 2015).
8. Andersen I. DNV GL vil ha ubemannede, flytende LNG-anlegg [DNV GL wants unmanned floating LNG facilities]. *Tekniske Ukeblad*. Tekniske Ukeblad Media AS 15 February 2015, http://www.tu.no/petroleum/2015/02/15/dnv-gl-vil-ha-ubemannede-flytende-lng-anlegg (accessed 23 July 2015).
9. Norwegian Maritime Authority. World's first test area for autonomous ships opened, 2016, https://www.sjo

fartsdir.no/en/news/news-from-the-nma/worlds-first-test-area-for-autonomous-ships-opened/ (accessed 7 October 2016).

10. Maritime Unmanned Navigation through Intelligence in Networks (MUNIN), 2012, http://www.unmanned-ship.org/munin/ (accessed 23 July 2016).
11. AAWA. Remote and autonomous ships—the next steps. In: Laurinen M (ed.) *Advanced autonomous waterborne applications*. London: AAWA, 2016, p.88.
12. Bertram V. *Unmanned surface vehicles—a survey*. Copenhagen: Skibsteknisk Selskab, 2008, pp.1–14.
13. Manley JE. Unmanned surface vehicles, 15 years of development. In: *Proceedings of the OCEANS 2008*, Quebec City, QC, Canada, 15–18 September 2008, vols 1–4, pp.1707–1710. New York: IEEE.
14. Yan R, Pang S, Sun H, et al. Development and missions of unmanned surface vehicle. *J Mar Sci Appl* 2010; 9: 451–457.
15. Yuh J, Marani G and Blidberg DR. Applications of marine robotic vehicles. *Intel Serv Robot* 2011; 4: 221–231.
16. Niu H, Adams S, Lee K, et al. Applications of autonomous underwater vehicles in offshore petroleum industry environmental effects monitoring. *J Can Petrol Technol* 2009; 48: 12–16.
17. Haugen J, Imsland L, Løset S, et al. Ice observer system for ice management operations. In: *Proceedings of the 21st international offshore (Ocean) and polar engineering conference* (eds JS Chung, SY Hong, I Langen, et al.), Maui, HI, 19–24 June 2011, pp.1120–1127. Cupertino, CA: ISOPE.
18. Griffiths G, Bose N, Ferguson J, et al. Insurance for autonomous underwater vehicles. *Underwater Technol* 2007; 27: 43–48.
19. Harris CA, Phillips AB, Dopico-Gonzalez C, et al. Risk and reliability modelling for multi-vehicle marine domains. In: *Proceedings of the 2016 IEEE/OES autonomous underwater vehicles (AUV)*, Tokyo, Japan, 6–9 November 2016, pp.286–293. New York: IEEE.
20. Vagia M, Transeth AA and Fjerdingen SA. A literature review on the levels of automation during the years. What are the different taxonomies that have been proposed? *Appl Ergon* 2016; 53(part A): 190–202.
21. Insaurralde CC and Lane DM. Autonomy-assessment criteria for underwater vehicles. In: *Proceedings of the 2012 IEEE/OES autonomous underwater vehicles (AUV)*, Southampton, 24–27 September 2012, pp.1–8. New York: IEEE.
22. Wang YC and Liu JG. Evaluation methods for the autonomy of unmanned systems. *Chinese Sci Bull* 2012; 57: 3409–3418.
23. Kirkwood WJ. AUV incidents and outcomes. In: *Proceedings of the MTS/IEEE Biloxi—marine technology for our future: global and local challenges OCEANS 2009*, Biloxi, MS, 26–29 October 2009, pp.1–5. New York: IEEE.
24. Brito MP and Griffiths G. A Markov chain state transition approach to establishing critical phases for AUV reliability. *IEEE J Oceanic Eng* 2011; 36: 139–149.
25. Utne IB and Schjølberg I. A systematic approach to risk assessment: focusing on autonomous underwater vehicles and operations in Arctic areas. In: *Proceedings of the ASME 2014 33rd international conference on ocean, offshore and arctic engineering*, San Francisco, CA, 8–13 June 2014. New York: ASME.
26. Ho G, Pavlovic N and Arrabito R. Human factors issues with operating unmanned underwater vehicles. *Proc Hum Factors Ergon Soc Annu Meet* 2011; 55: 429–433.
27. Griffiths G, Millard NW, McPhail SD, et al. On the reliability of the Autosub autonomous underwater vehicle. *Underwater Technol* 2003; 25: 175–184.
28. Griffiths G and Brito M. Predicting risk in missions under sea ice with autonomous underwater vehicles. In: *Proceedings of the 2008 IEEE/OES autonomous underwater vehicles (AUV 2008)*, Woods Hole, MA, 13–14 October 2008, pp.1–7. New York: IEEE.
29. Brito MP and Griffiths G. Results of expert judgments on the faults and risks with Autosub3 and an analysis of its campaign to Pine Island Bay, Antarctica, 2009. In: *Proceedings of the international symposium on unmanned untethered submersible technology (UUST 2009)*, Durham, NH, 23–26 August 2009, p.14. Lee, NH: Autonomous Undersea Systems Institute (AUSI).
30. Brito MP, Griffiths G and Challenor P. Risk analysis for autonomous underwater vehicle operations in extreme environments. *Risk Anal* 2010; 30: 1771–1788.
31. Brito M and Griffiths G. A Bayesian approach for predicting risk of autonomous underwater vehicle loss during their missions. *Reliab Eng Syst Safe* 2016; 146: 55–67.
32. Griffiths G, Brito M, Robbins I, et al. Reliability of two REMUS-100 AUVs based on fault log analysis and elicited expert judgment. In: *Proceedings of the international symposium on unmanned untethered submersible technology (UUST 2009)*, Durham, NH, 23–26 August 2009, p.12. Durham, NH: Autonomous Undersea Systems Institute (AUSI).
33. Rødseth ØJ and TjoraÅ. A risk based approach to the design of unmanned ship control systems. In: S Ehlers, BE Asbjornslett, ØJ Rødseth, et al. (eds) *Maritime-port technology and development*. Boca Raton, FL: CRC Press, 2014, pp.153–161.
34. Rødseth ØJ and Burmeister H-C. Risk assessment for an unmanned merchant ship. *TransNav* 2015; 9: 357–364.
35. Kretschmann L, Rødseth ØJ, TjoraÅ, et al. D9.2: qualitative assessment. In: *Maritime unmanned navigation through intelligence in networks*. 1st ed. 2015, p.45, http://www.unmanned-ship.org/munin/wp-content/uploads/2015/10/MUNIN-D9-2-Qualitative-assessment-CML-final.pdf
36. Kretschmann L, Rødseth ØJ, Fuller BS, et al. D9.3: quantitative assessment. In: *Maritime unmanned navigation through intelligence in networks*, 2015, http://www.unmanned-ship.org/munin/wp-content/uploads/2015/10/MUNIN-D9-3-Quantitative-assessment-CML-final.pdf
37. Ono M, Quadrelli M and Huntsberger TL. Safe maritime autonomous path planning in a high sea state. In: *Proceedings of the 2014 American control conference (ACC 2014)*, Portland, OR, 4–6 June 2014, pp.4727–4734. Piscataway, NJ: IEEE.
38. Li Z, Bachmayer R and Vardy A. Risk analysis of an autonomous surface craft for operation in harsh ocean environments. In: *Proceedings of the 2016 IEEE/OES autonomous underwater vehicles (AUV 2016)*, Tokyo, Japan, 6–9 November 2016, pp.294–300. New York: IEEE.
39. Thieme CA, Utne IB and Schjølberg I. A risk management framework for unmanned underwater vehicles focusing on human and organizational factors. In: *Proceedings of the ASME 2015 34th international conference on ocean, offshore and arctic engineering (OMAE2015)*,

St. John's, NL, Canada, 31 May–5 June 2015. New York: ASME.

40. Thieme CA, Utne IB and Schjølberg I. Risk modeling of autonomous underwater vehicle operation focusing on the human operator. In: *Proceedings of the 25th European safety and reliability conference (ESREL 2015)* (ed L Podofillini, B Sudret, B Stojadinovic, et al), Zürich, 7–10 September 2015, pp.3653–3660. Boca Raton, FL: CRC Press, Taylor & Francis Group.
41. Cummings M. Man versus machine or man plus machine? *IEEE Intell Syst* 2014; 29: 62–69.
42. Hollnagel E. Human reliability assessment in context. *Nucl Eng Technol* 2005; 37: 159–166.
43. Kjærulff UB and Madsen AL. *Bayesian networks and influence diagrams: a guide to construction and analysis*. 2nd ed. New York: Springer Science + Business Media, 2013.
44. Martins MR and Maturana MC. The application of the Bayesian networks in the human reliability analysis. In: *Proceedings of the ASME 2009 international mechanical engineering congress and exposition*, Lake Buena Vista, FL, 13–19 November 2009, pp.341–348. New York: IEEE.
45. Akhtar MJ and Utne IB. Human fatigue's effect on the risk of maritime groundings—a Bayesian network modeling approach. *Safety Sci* 2014; 62: 427–440.
46. Mazaheri A, Montewka J and Kujala P. Towards an evidence-based probabilistic risk model for ship-grounding accidents. *Safety Sci* 2016; 86: 195–210.
47. Vinnem JE, Bye R, Gran BA, et al. Risk modelling of maintenance work on major process equipment on offshore petroleum installations. *J Loss Prevent Proc* 2012; 25: 274–292.
48. Gran BA, Bye R, Nyheim OM, et al. Evaluation of the Risk OMT model for maintenance work on major offshore process equipment. *J Loss Prevent Proc* 2012; 25: 582–593.
49. Trucco P, Cagno E, Ruggeri F, et al. A Bayesian belief network modelling of organisational factors in risk analysis: a case study in maritime transportation. *Reliab Eng Syst Safe* 2008; 93: 845–856.
50. Jensen FV and Nielsen TD. *Bayesian networks and decision graphs*. New York: Springer Science & Business Media, 2009.
51. Decision Systems Laboratory of the University of Pittsburgh. *GeNIe modeling environment* (GeNIe software 2.0). Pittsburgh, PA: University of Pittsburgh, 2013.
52. McKendrick R, Shaw T, Saqer H, et al. Team performance and communication within networked supervisory control human-machine systems. *Proc Hum Factors Ergon Soc Annu Meet* 2011; 55: 262–266.
53. Chen JYC, Barnes MJ and Harper-Sciarini M. Supervisory control of multiple robots: human-performance issues and user-interface design. *IEEE T Syst Man Cy C* 2011; 41: 435–454.
54. Fouse S, Champion M and Cooke NJ. The effects of vehicle number and function on performance and workload in human-robot teaming. *Proc Hum Factors Ergon Soc Annu Meet* 2012; 56: 398–402.
55. Parasuraman R and Miller CA. Trust and etiquette in high-criticality automated systems. *Commun ACM* 2004; 47: 51–55.
56. Parasuraman R and Riley V. Humans and automation: use, misuse, disuse, abuse. *Hum Factors* 1997; 39: 230–253.
57. Sheridan TB and Parasuraman R. Human-automation interaction. *Rev Hum Fact Ergon* 2005; 1: 89–129.
58. Wiener EL and Curry RE. Flight-deck automation: promises and problems. *Ergonomics* 1980; 23: 995–1011.
59. Johnson RC, Saboe KN, Prewett MS, et al. Autonomy and automation reliability in human-robot interaction: a qualitative review. In: *Proceedings of the 53rd human factors and ergonomics society annual meeting 2009 (HFES 2009)*, San Antonio, TX, 19–23 October 2009, pp.1398–1402. Santa Monica, CA: Human Factors an Ergonomics Society.
60. Gander P, Hartley L, Powell D, et al. Fatigue risk management: organizational factors at the regulatory and industry/company level. *Accident Anal Prev* 2011; 43: 573–590.
61. Fincannon T, Jentsch F, Sellers B, et al. Best practices in human operation of robotic/unmanned vehicles: a technical review of recommendations regarding the human-to-robot ratio. In: *Proceedings of the 57th human factors and ergonomics society annual meeting (HFES 2013)*, San Diego, CA, 30 September–4 October 2013, pp.1268–1272. Santa Monica, CA: Human Factors an Ergonomics Society.
62. De Visser E, Shaw T, Mohamed-Ameen A, et al. Modeling human-automation team performance in networked systems: individual differences in working memory count. *Proc Hum Factors Ergon Soc Annu Meet* 2010; 54: 1087–1091.
63. Baxter GD and Bass EJ. Human error revisited: some lessons for situation awareness. In: *Proceedings of the 1998 4th annual symposium on human interaction with complex systems*, Dayton, OH, 22–25 March 1998, pp.81–87. New York: IEEE.
64. Ruff HA, Narayanan S and Draper MH. Human interaction with levels of automation and decision-aid fidelity in the supervisory control of multiple simulated unmanned air vehicles. *Presence: Teleop Virt* 2002; 11: 335–351.
65. Lee JD and See KA. Trust in automation: designing for appropriate reliance. *Hum Factors* 2004; 46: 50–80.
66. Endsley MR and Kaber DB. Level of automation effects on performance, situation awareness and workload in a dynamic control task. *Ergonomics* 1999; 42: 462–492.
67. Schuster D, Jentsch F, Fincannon T, et al. The impact of type and level of automation on situation awareness and performance in human-robot interaction. In: Harris D (ed.) *Engineering psychology and cognitive ergonomics understanding human cognition*. Berlin; Heidelberg: Springer, 2013, pp.252–260.
68. Groth KM and Swiler LP. Bridging the gap between HRA research and HRA practice: a Bayesian network version of SPAR-H. *Reliab Eng Syst Safe* 2013; 115: 33–42.
69. Garcia JC, Fernandez JJ, Sanz PJ, et al. Increasing autonomy within underwater intervention scenarios: the user interface approach. In: *Proceedings of the 2010 4th annual IEEE systems conference*, San Diego, CA, 5–8 April 2010, pp.71–75. New York: IEEE.
70. Donmez B, Nehme C and Cummings ML. Modeling workload impact in multiple unmanned vehicle supervisory control. *IEEE T Syst Man Cy A* 2010; 40: 1180–1190.

71. Squire PN and Parasuraman R. Effects of automation and task load on task switching during human supervision of multiple semi-autonomous robots in a dynamic environment. *Ergonomics* 2010; 53: 951–961.
72. Oakley B, Mouloua M and Hancock P. Effects of automation reliability on human monitoring performance. *Proc Hum Factors Ergon Soc Annu Meet* 2003; 47: 188–190.
73. Cummings ML, Clare A and Hart C. The role of human-automation consensus in multiple unmanned vehicle scheduling. *Hum Factors* 2010; 52: 17–27.
74. Cummings ML, Bertucelli LF, Macbeth J, et al. Task versus vehicle-based control paradigms in multiple unmanned vehicle supervision by a single operator. *IEEE T Hum-Mach Syst* 2014; 44: 353–361.
75. Giese S, Carr D and Chahl J. Implications for unmanned systems research of military UAV mishap statistics. In: *Proceedings of the IEEE intelligent vehicles symposium*, Gold Coast, QLD, Australia, 23–26 June 2013, pp.1191–1196. New York: IEEE.
76. Endsley MR. Toward a theory of situation awareness in dynamic-systems. *Hum Factors* 1995; 37: 32–64.
77. Gehrke JD. Evaluating situation awareness of autonomous systems. In: Madhavan R, Tunstel E and Messina E (eds) *Performance evaluation and benchmarking of intelligent systems*. New York: Springer, 2009, pp.93–111.
78. Riley V. A general model of mixed-initiative human-machine systems. In: *Proceedings of the human factors society 33rd annual meeting perspectives*, Denver, CO, 16–20 October 1989, pp.124–128. Santa Monica, CA: Human Factors Society.
79. Hanninen M. Bayesian networks for maritime traffic accident prevention: benefits and challenges. *Accident Anal Prev* 2014; 73: 305–312.
80. Mkrtchyan L, Podofillini L and Dang VN. A survey of Bayesian belief network applications in human reliability analysis. In: Nowakowski T, Młyńczak M, Jodejko-Pietruczuk A, et al. (eds) *Safety and reliability: methodology and applications—proceedings of the European safety and reliability conference (ESREL 2014)*. Boca Raton, FL: CRC Press, 2015, pp.1073–1081.
81. Gertman D, Blackman H, Marble J, et al. *The SPAR-H human-reliability analysis method*. NUREG/CR-6883, August 2005. Washington, DC: NUREG, U.S. Nuclear Regulatory Commission.
82. Strutt JE. *Report of the inquiry into the loss of Autosub2 under the Fimbulisen*. Southampton: National Oceanography Centre Southampton, 2006.
83. Mkrtchyan L, Podofillini L and Dang VN. Overview of methods to build conditional probability tables with partial expert information for Bayesian belief networks. In: Podofillini L, Sudret B, Stojadinovic B, et al. (eds) *Safety and reliability of complex engineered systems—proceedings of the 25th European safety and reliability conference (ESREL 2015)*. Boca Raton, FL: CRC Press, 2015, pp.1973–1981.
84. Pitchforth J and Mengersen K. A proposed validation framework for expert elicited Bayesian Networks. *Expert Syst Appl* 2013; 40: 162–167.
85. Pitchforth J, Wu P and Mengersen K. Applying a validation framework to a working airport terminal model. *Expert Syst Appl* 2014; 41: 4388–4400.
86. The Applied Underwater Robotics Laboratory (AUR Lab), 2015, http://www.ntnu.edu/aur-lab (accessed 2 February 2015).
87. Thieme CA. *Development of a risk management process for NTNU's REMUS 100 AUV*. Master's Thesis, Department of Marine Technology, Norwegian University of Science and Technology, Trondheim, 2014, p.105.
88. Hydroid. Remus 100-autonomous underwater vehicle. Hydroid—a Kongsberg group, 2016, https://www.km.kongsberg.com/ks/web/nokbg0240.nsf/AllWeb/D241A2C835DF40B0C12574AB003EA6AB?OpenDocument
89. Hagen PE, Hegrenæs Ø, Jalving B, et al. Making AUVs truly autonomous. In: Inzartsev AV (ed.) *Underwater vehicles*. Vienna: InTech, 2009, p.582.
90. Ho G, Pavlovic NJ, Arrabito R, et al. *Human factors issues when operating unmanned underwater vehicles*. Ottawa, ON, Canada: Defence R&D Canada, 2011.
91. Parasuraman R and Wickens CD. Humans: still vital after all these years of automation. *Hum Factors* 2008; 50: 511–520.
92. Thieme CA and Utne IB. Safety performance monitoring of autonomous marine systems. *Reliab Eng Syst Safe* 2017; 159: 264–275.
93. Hart SG and Staveland LE. Development of NASA-TLX (Task Load Index): results of empirical and theoretical research. In: Hancock PA and Meshkati N (eds) *Human mental workload*. Amsterdam: Elsevier, 1988, pp.139–183.
94. Kozine I. Simulation of human performance in time-pressured scenarios. *Proc IMechE, Part O: J Risk and Reliability* 2007; 221: 141–151.

Appendix 1

Assessment of influence of strength for CPT building

This section summarizes the considerations underlying the CPT assessment. For each child node, except for Autonomous Function Performance and HAC, which are in the main body of this article, the parent nodes, their influence and associated considerations are presented in Tables 8–14. The assessment was conducted by the authors and supported with input from the literature, as indicated. The assessment was conducted for AUV-specific operation

Table 8. Strength rating and associated reasoning for the CPT fatigue, these considerations are supported by results of Akhtar and Utne.[45].

Parent state	Strength	Reasoning
Mission Duration	Low	The mission duration has a low influence on fatigue, since the operators will still have to fulfil their shift lengths. Shorter missions will give more room for short breaks and hence, only have little effect
Shift Scheme	High	Insufficient length of rest and sleep can lead to strong effects of fatigue
Workload	High	Workload influences fatigue strongly, since it represents the cognitive work and the exhaustion of these capabilities

Table 9. Strength rating and associated reasoning for the CPT feedback from the system.

Parent state	Strength	Reasoning
Etiquette	High	Research shows that the way information is presented has a significant influence on the operator[55]
False Alarm Rate	Low	In comparison to Etiquette and information presentation, the False Alarm Rate has only a marginal influence on the operator[55]
Interface Design	High	The quality of interfaces, both physical and virtual, highly influences the way information is perceived[57]

Table 10. Strength rating and associated reasoning for the CPT human operator performance.

Parent state	Strength	Reasoning
Fatigue	Low	Fatigue is seen as a contributing factor to the performance of operators, not as a decisive factor. A fatigued operator can still perform adequately. Additionally, the role of fatigue in AUV operation and human–autonomy collaboration is not well analyzed, and the role of fatigue shall not be overemphasized
Operators' Experience	High	Operators' Experience is highly important, in order to perform their tasks. It enables them to operate the system efficiently
Operators' Training	High	Operators' Training is highly important, in order to perform their tasks. It enables them to take the right actions
Procedures	Low	It is believed that procedures have a low influence, in order to reflect that for normal operation they are important, but have limited influence in critical situations
Reaction Time	Low	The Reaction Time is of low influence. AUVs are rather slow and most situations leave a sufficient long time to react
SA of Human Operators	High	SA of Human Operators is highly influential, since it determines the operators' operational picture of the AUV mission. This is a decisive factor, for the operators to know what to do

Table 11. Strength rating and associated reasoning for the CPT reaction time.

Parent state	Strength	Reasoning
Operators' Experience	High	Experience improves reaction time
Operators' Training	Low	The influence of training was assumed low, since it implies to implement the right actions timely. However, training, in the sense of courses and workshops, only addresses this issue in a limited way
Time Delay of Transmission	High	Status messages and commands travel relative slowly through water. Hence, the Reaction Time is highly dependent on the delay of important commands send to the AUVs or messages received from the AUVs
Workload	High	Occupation with other tasks, especially complex ones, has proven to increase the operators' time to switch to another task that needs attention[71]

Table 12. Strength rating and associated reasoning for the CPT SA of human operators.

Parent state	Strength	Reasoning
Communication	Low	Information is mainly communicated through interface of the system. Hence, the influence is assumed low
Feedback from the System	High	Feedback from the System is highly important for the operators[55]
Operators' Training	High	Training of the operators is highly important for the operators to create an operational picture of the current operation
Time Delay of Transmission	Low	The delay of information updating reduces the knowledge about the current state of a mission. Since, no video streams or direct control are possible in current AUV operation,[26] it was assumed low
Trust	High	Inadequate Trust in a system is decisive for SA of Human Operators[56]
Workload	High	A high Workload of the operators has been shown to reduce SA of Human Operators significantly[64]

Table 13. Strength rating and associated reasoning for the CPT Trust.

Parent state	Strength	Reasoning
Feedback from the System	High	The way a system presents information is highly important for building an adequate level of trust[55,57]
Operators' Experience	High	Experience with a system builds Trust.[65] Hence, a high influence is assumed
Operators' Training	Low	Training can give understanding for the system, guidance in usage and handling of systems. However, training will only make a system more trustable.[65] Hence, it is assumed to have a low influence
Reliability of Autonomous Functions	High	The influence of Reliability of Autonomous Functions is high. People tend to project emotions on systems. Reliable systems are easily trusted[65]

Table 14. Strength rating and associated reasoning for the CPT workload.

Parent state	Strength	Reasoning
LOA	Low	The LOA has only a marginal influence on the operator Workload.[66] it is believed that the same is true for AUV operation
Task Load	High	Carrying out tasks concurrently will increase the workload highly
Number of Vehicles per Operator	High	The number of vehicles effectively increases the number of tasks[62]

This page is intentionally left blank

Article 5

Thieme, C. A. & Utne, I. B. 2017. *Safety performance monitoring of autonomous marine systems*. *Reliability Engineering & System Safety,* 159*,* March, pp. 264-275, DOI: 10.1016/j.ress.2016.11.024

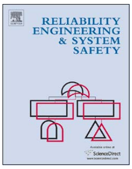
This page is intentionally left blank

Reliability Engineering and System Safety 159 (2017) 264–275

Contents lists available at ScienceDirect

Reliability Engineering and System Safety

journal homepage: www.elsevier.com/locate/ress



Safety performance monitoring of autonomous marine systems

Christoph A. Thieme[a,*], Ingrid B. Utne[b]

[a] Centre for Autonomous Marine Operations and Systems (AMOS), Department of Marine Technology, Norwegian University of Science and Technology Trondheim (NTNU), Norway
[b] Department of Marine Technology, NTNU Trondheim, Norway

ARTICLE INFO

Keywords:
Safety indicators
Autonomous marine systems
Dual assurance
Resilience

ABSTRACT

The marine environment is vast, harsh, and challenging. Unanticipated faults and events might lead to loss of vessels, transported goods, collected scientific data, and business reputation. Hence, systems have to be in place that monitor the safety performance of operation and indicate if it drifts into an intolerable safety level. This article proposes a process for developing safety indicators for the operation of autonomous marine systems (AMS). The condition of safety barriers and resilience engineering form the basis for the development of safety indicators, synthesizing and further adjusting the dual assurance and the resilience based early warning indicator (REWI) approaches. The article locates the process for developing safety indicators in the system life cycle emphasizing a timely implementation of the safety indicators. The resulting safety indicators reflect safety in AMS operation and can assist in planning of operations, in daily operational decision-making, and identification of improvements. Operation of an autonomous underwater vehicle (AUV) exemplifies the process for developing safety indicators and their implementation. The case study shows that the proposed process leads to a comprehensive set of safety indicators. It is expected that application of the resulting safety indicators consequently will contribute to safer operation of current and future AMS.

1. Introduction

Marine systems are becoming more automated and autonomous, with increasing technological complexity. In the future, autonomous marine systems (AMS), such as unmanned surface vessels, autonomous underwater vehicles (AUV), and other types of underwater robots will lead to improved maritime transportation, research of the oceans and arctic regions, military operations, and inspection and maintenance of subsea hydrocarbon production facilities [16,31–33,52,53,59]. This development is accelerated by the pressure to reduce costs, risks, and a demand for achieving more environmental friendly and sustainable operation.

Autonomy is a system's ability to make decisions, in order to fulfill a task, without the need for assistance of an operator or external agent during task performance [55]. An AMS is therefore not necessarily unmanned. The level of autonomy describes the degree and extent of decision-making, problem solving and strategy implementation of the system, when faced with uncertainty or unanticipated events [23]. Scales, e.g., from 1 to 10, for the level of autonomy range from manual control to full autonomy, of which the latter means no possibility for intervention from the operator. Levels in between include, for example, decision making by humans and implementation by the system, so called batch processing; shared plan generation and execution of tasks, where the operators still have full decision authority, so called shared control; and plan generation and execution by the system, where the operators only intervene if necessary, so called supervisory control [12]. Vagia et al. [55] give a comprehensive overview of different scales for levels of autonomy proposed in the literature. Not every AMS has the same level of autonomy in every subsystem or for each capability. For example, Insaurralde and Lane [23] differentiate between different problem-solving capabilities and the context for which the AUV is considered. Current AMS are not fully autonomous as they are supervised, with different ways of intervention from the operators, or they are remotely operated [38].

AMS can be operated with few or no human operators on board, which may decrease the risk of operation in relation to crew injuries and fatalities. Remote supervision and control, however, create risk in relation to other marine stakeholders, material assets, and the environment. During critical situations, which the AMS may not be capable of handling, operators have to take control and identify the right course of action, to avoid a potential incident. This requires high situation awareness of the operators and adequate input from support systems to handle such situations [3,38]. Additional challenges are created by human interaction with the system during design, maintenance, or

[*] Corresponding author.
E-mail address: Christoph.thieme@ntnu.no (C.A. Thieme).

http://dx.doi.org/10.1016/j.ress.2016.11.024
Received 8 March 2016; Received in revised form 15 October 2016; Accepted 27 November 2016
Available online 28 November 2016

definition of overall mission goals [18]. The influence of the organization operating AMS is not negligible and has to be considered sufficiently during development and use.

Few publications cover risk in relation to AMS. Most of them focus on AUV, e.g., risk management [5,50,54], risk assessments [6–8,13,14], incident investigation [30,47], or the influence of the human operators [17,49]. Unmanned and autonomous ships are briefly analyzed [37,38,43,44]. Huang et al. [21] propose a generic framework for deriving contextual performance metrics for unmanned systems, but do not cover safety, explicitly. In general, risk assessments and hazard identification should be reviewed, regularly [24]. Currently, review and subsequent updating, however, may be carried out after several years in operation. Changes in environmental, technical and organizational conditions may occur in shorter intervals than the reviews [27]. Hence, there is a need for indicators to measure safety performance and methods for analysis and monitoring of risk and safety during operation of AMS.

The objective of this article is to propose a structured process for developing safety indicators for AMS to be used for monitoring the operational safety performance of AMS. The methodological approach in the article is based on safety indicator development processes from high-risk industries, which are adjusted to the context of AMS. The feasibility and usefulness of the process is demonstrated for an AUV. The proposed safety indicators are evaluated for applicability in operational decision-making and safety monitoring. The process for developing safety indicators in this article addresses a company and system level, which means that an industrial or global industry scale are outside the scope, although some indicators might be also applicable on such a high level.

The next Section discusses the concepts of risk and safety indicators and methods for their development. This is followed by the description of a synthesized process for developing safety indicators based on the reviewed methods. Section 4 exemplifies the proposed process for developing safety indicators and presents safety indicators for an AUV. The last Section discusses and concludes the presented work.

2. Safety indicators

High-risk industries use risk and safety indicators to monitor the status of major hazards at an industrial level, e.g. [56], at a company level, e.g. [41], or at a single plant or unit of operation, e.g. [15,46,68]. Risk and safety indicators are specific for a certain organizational level. Indicators aiming at an industrial level might not be applicable to only one company or one specific plant.

Different definitions of risk and safety indicators are in use. Although used similarly and sometimes synonymously, risk and safety indicators are not the same. Risk indicators are derived from a risk based approach [64], e.g. [60,61]. A risk indicator is the operational measurable variable related to a risk-influencing factor (RIF) in a risk model [64]. This article focuses on safety indicators. Safety is a condition where the remaining risk is accepted as sufficiently low [39], and safety indicators measure to which extent safety is present. Safety indicators include event indicators, barrier indicators, activity indicators, and programmatic indicators [68]. Øien [61] defines an indicator generally as “a measurable or operational variable that can be used to describe the condition of a broader phenomenon or aspect of reality”. Here, the condition of a broader phenomenon is the level of safety in operation. Hence, a safety indicator is a measurable or an operational variable that can be used to describe the level of safety of operation. Swuste et al. [48] present and discuss other definitions in use in the scientific community and in different industries.

Two main types of safety indicators exist; occupational safety indicators and process safety indicators. Past accidents show that occupational safety indicators only cannot be used to monitor changes in process risk [15,19,27,65], such as the Macondo Blowout in 2010 [10]. In this article, occupational safety indicators are excluded from further consideration, since few or no personnel will be on board the AMS during operation in the future.

Many safety indicator approaches distinguish between leading and lagging indicators. Hopkins [19] discusses the meaning and usefulness of this distinction. Essentially, a leading indicator indicates if the safety level of an organization is changing. However, actions can still be taken to avoid an accident [11,26]. Lagging indicators include events that are considered an accident or incident. Leading and lagging indicators can be ambiguous terms [11,19]. Hence, in this article, the terms “early warning” and “outcome” indicators are used in the context of AMS safety indicators, instead of leading and lagging indicators, in attempt to reduce any confusion. Early warning indicators provide information on an unsatisfactory performance of a safety barrier, related to preventing a potential incident [62]. Safety barriers can be physical or engineered systems, as well as human actions, which are guided by procedures or organizational initiatives. These shall prevent, control or mitigate harm from hazards [39]. An outcome indicator is an indicator related to the manifestation of undesired events. These reflect actual operational safety performance [22].

Different safety indicators consider different periods of change, since some changes occur slower than others [27]. Hence, efforts are made to capture fast changing safety factors, to include them in real-time safety monitoring, e.g., by Knegtering and Pasman [27], or Vinnem et al. [57].

To select an appropriate, complementary and manageable set of safety indicators, the proposed safety indicators have to be evaluated against a set of required characteristics [20,22,25,26,60,65,66]. Table 1 summarizes the characteristics from [22,25,26,60], which are found particularly relevant for AMS. These will be used throughout this article.

2.1. Safety indicator development methods

Delatour et al. [9] and Øien [68] review and discuss methods for safety performance indicator development. Leveson [28] sets requirements for a good leading indicator development process. In short, it should be complete, consistent, effective, traceable, minimal, continually improving, and unbiased.

Two indicator development methods are found most suitable for further development and adjustment to the context of AMS; the dual assurance method [20], and the resilience based early warning indicators (REWI) method [66]. The dual assurance method provides an overview of the performance of important safety barriers. Especially, technical safety barriers, such as, sensor systems and collision avoidance systems, are relevant for AMS, since they give relevant input to the control system of the AMS and its operators. Furthermore, the method is a practical approach for safety indicators and widely accepted and used in the process industry [36]. However, other industries, which require a high level of confidence in their systems operating correctly and safely, can apply the approach [20]. Other approaches, such as API RP 754 [1], OECD Guidance No. 19 [34] and OGP Report No. 456 [35], are similar to the dual assurance method. However, they focus specifically on the release of hazardous materials, which is a more specific application area of less relevance for AMS.

In AMS operation, the operators have to be aware of the situation

Table 1
Selected safety indicator evaluation criteria, based on [22,25,26,60].

Safety indicator evaluation criteria	
1	Relationship between safety indicator and safety is evident and understood
2	The safety indicator is observable and sufficiently measurable
3	Data is already collected or can be collected
4	Measurements are repeatable and verifiable
5	The safety indicator is robust against manipulation

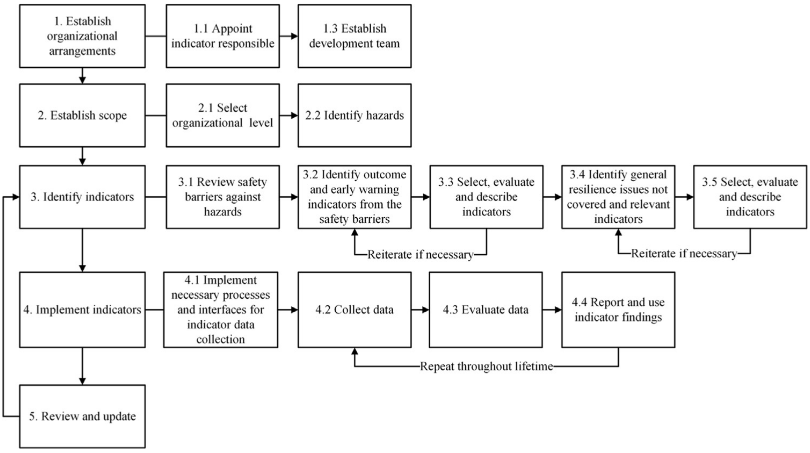

1. Establish organizational arrangements
1.1 Appoint indicator responsible
1.3 Establish development team
2. Establish scope
2.1 Select organizational level
2.2 Identify hazards
3. Identify indicators
3.1 Review safety barriers against hazards
3.2 Identify outcome and early warning indicators from the safety barriers
3.3 Select, evaluate and describe indicators
3.4 Identify general resilience issues not covered and relevant indicators
3.5 Select, evaluate and describe indicators
Reiterate if necessary
Reiterate if necessary
4. Implement indicators
4.1 Implement necessary processes and interfaces for indicator data collection
4.2 Collect data
4.3 Evaluate data
4.4 Report and use indicator findings
Repeat throughout lifetime
5. Review and update

Fig. 1. Steps in the synthesized process for developing safety indicators for AMS, based on [20,66].

and be able to make the right decisions in those cases, where the AMS reaches its operational limits [38]. Many AMS today are still in development, unique or built in small numbers. Therefore, limited operational experience exists with AMSs making it important to monitor the supporting organization to ensure appropriate operation. REWI focuses on organizational performance to handle accidents, incidents and unexpected events. It aims at management decisions, appropriate communication within an organization and risk management, which is highly relevant for AMS. According to Øien and Paltrinieri [67], the dual assurance and the REWI methods provide effective and complementary means for developing safety indicators.

2.1.1. *The dual assurance method*

The UK health and safety executive (HSE) [20] developed the dual assurance approach together with the chemical and major hazard processing industry. The method assists in establishing key performance indicators for major hazards and process safety. The dual assurance method employs leading and lagging indicators and compares the lagging indicators to the leading indicators to reveal if the measured safety performance reflects the actual safety performance [20]; i.e., dual assurance. Safety indicators originate from the risk control systems (RCS) [20]. Reason's [40] layers of defense form the basis of the method. Organizational accidents arise due to inadequacies in the RCS, which promote active failures, leading to accidents. The RCS should be part of a safety management system, which focuses on a specific risk or activity [20]. Examples are sensors and alarms, the permit to work system, inspection and maintenance.

The dual assurance method is to some extent generic, even though it is developed for a chemical process plant. Hence, methodological adaptations to AMS are necessary, such as:

- The steps of the dual assurance development process have to be rearranged in order to fit it to the AMS' lifecycle.
- The term safety barrier, more commonly used in the marine industry, replaces RCS of the dual assurance method.
- The dual assurance method does not include consideration in terms of sampling intervals of the safety indicators, but these need to be defined for prudent use of indicators.

2.1.2. *The resilience based early warning approach to development of indicators*

The REWI method [66] was developed to prevent major accidents and to improve organizational safety and performance. The method is an extension of the leading indicators of the organizational health method, proposed by the US electric power research institute [63]. Resilience thinking forms the basis for the REWI method. Woods [58] describes resilience as the ability to recognize and adapt to unexpected changes in operation, in order to handle such changes. Therefore, a resilient organization is one that monitors its ability to foresee, recognize, and handle unexpected changes, and adjusts if these competences are not satisfying a certain level [42].

REWI [66] applies contributing success factors (CSF), derived from the attributes of resilience (risk awareness, response capacity, support), to develop the safety indicators. The CSF are risk understanding, anticipation, attention, response, robustness, resourcefulness/ rapidity, decision support and redundancy [51]. General issues defined by Øien et al. [43] describe considerations and practices, which apply to most high-risk industries and are necessary to achieve the CSF for a resilient organization. For these general issues, REWI proposes a set of measurable safety indicators, but leaves room for adding or adapting general issues and safety indicators to suit the organization and operation.

The REWI method aims at determining the organizational capabilities to handle unexpected and undesired situations, which might result in an accident. These are important aspects for the operation of AMS. Operators have to be prepared to make the right decisions and actions in case of failing systems. Especially, the CSF attention, response, resourcefulness/ rapidity and decision support are key factors in operation of AMS. Most of the general issues suggested in REWI are relevant for operation of AMS. Depending on its operating organization and its practices, other general issues and associated indicators may be necessary to identify.

By synthesizing the dual assurance and the REWI approaches, synergy effects are expected compared to applying the development processes individually. The expected benefits are reduced use of resources and time for identification of indicators, and a more adequate and comprehensive set of safety indicators. The resilience indicator process focuses on the CSF that are not covered sufficiently by the safety indicators related to safety barriers (dual assurance).

3. A process for developing safety indicators for autonomous marine systems

Fig. 1 presents an overview of the proposed process for developing safety indicators, with five main steps and several sub steps. Detailed descriptions of each step follow in the next sub Sections.

Fig. 2 shows how the process for developing safety indicators relates to the life cycle phases of AMS, adapted from Blanchard [4].

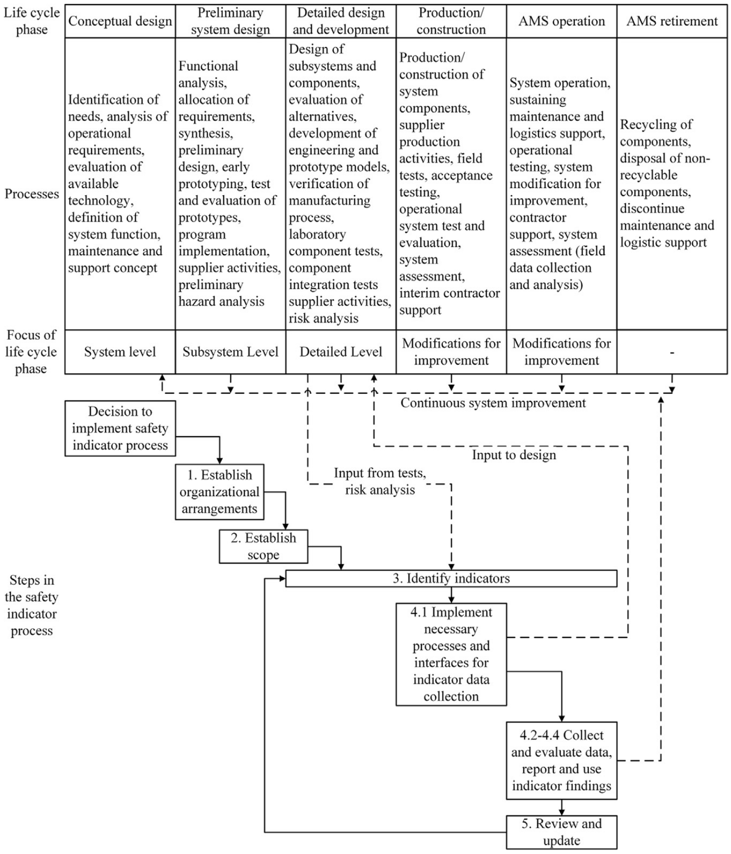

Life cycle phase
Conceptual design
Preliminary system design
Detailed design and development
Production/ construction
AMS operation
AMS retirement
Processes
Identification of needs, analysis of operational requirements, evaluation of available technology, definition of system function, maintenance and support concept
Functional analysis, allocation of requirements, synthesis, preliminary design, early prototyping, test and evaluation of prototypes, program implementation, supplier activities, preliminary hazard analysis
Design of subsystems and components, evaluation of alternatives, development of engineering and prototype models, verification of manufacturing process, laboratory component tests, component integration tests supplier activities, risk analysis
Production/ construction of system components, supplier production activities, field tests, acceptance testing, operational system test and evaluation, system assessment, interim contractor support
System operation, sustaining maintenance and logistics support, operational testing, system modification for improvement, contractor support, system assessment (field data collection and analysis)
Recycling of components, disposal of non-recyclable components, discontinue maintenance and logistic support
Focus of life cycle phase
System level
Subsystem Level
Detailed Level
Modifications for improvement
Modifications for improvement
-
Continuous system improvement
Decision to implement safety indicator process
1. Establish organizational arrangements
Input to design
Input from tests, risk analysis
2. Establish scope
Steps in the safety indicator process
3. Identify indicators
4.1 Implement necessary processes and interfaces for indicator data collection
4.2-4.4 Collect and evaluate data, report and use indicator findings
5. Review and update

Fig. 2. The process for developing safety indicators in relation to the system life cycle of an AMS (system lifecycle adapted from Blanchard [4], figure 1.12). Solid lines represent the sequential order of steps. Dashed lines represent feedback and feedforward of information and initiation of reiterations.

The Figure includes the development, operational, and improvement processes that are undertaken during the major life cycle phases, the phases of the process for developing safety indicators and the feedback and input from the different phases and activities (dashed lines). The life cycle of AMS is divided into six phases, characterized by an initial top down approach starting at the system level in the conceptual design phase. Through the preliminary design and development phase, the focus gradually narrows down to the component and detailed design level, initiating a bottom up approach ending with system integration, testing and verification, before and during the commissioning of the AMS. The combined top-down and bottom-up approaches constitute the Vee model [4]. For efficient development and implementation, the process for developing safety indicators should start during the conceptual AMS design and progress as the system evolves and reaches its operational phase.

3.1. Establish organizational arrangements

A successful safety indicator system requires the commitment and trust of the management, in order to get required resources and support for development and use of the indicator system. The decision to implement a process for developing safety indicators for an AMS should be made during the conceptual design phase. Organizational arrangements are established during the preliminary design phase. A responsible for the indicator system should be appointed with support from management. He or she is responsible for organizing indicator development workshops, documentation of the safety indicators, indicator evaluation and presentation of indicator monitoring reports. The indicator system responsible appoints and commits the development team [66]. A development team should consist of four to eight people, including personnel who work in maintenance, operation, safety, and management. It might be beneficial to involve control and autonomy experts. Additionally, a secretary and a facilitator or mediator, to guide the indicator development workshops, are recommended. In the first indicator development workshop, the development team has to be introduced to safety indicators in general, common terminology, and the system itself [66]. Indicator development during the system design process has to utilize operational experience from operators with other similar existing systems, as well as qualified information and knowledge from the operators' point of view.

3.2. Establish scope

The second step is to establish the scope of the safety indicator system. This should occur during the preliminary design phase of the AMS and be finished with the beginning of detailed design and development, in order to ensure that meaningful safety indicators are identified and that the necessary interfaces for data collection are implemented timely in the system. The scope includes a description of the AMS, the organizational level the indicators aim at, major hazards, associated safety barriers, and their safe operational limits. In the context of AMS, the focus of the indicator system could be on one vessel, a fleet of vessels, the control center, or the company. For AMS, the major hazards are loss of AMS, or collision of AMS with other vessels or structures. The documentation of the scope should contain scenario descriptions and identification of underlying causes [20]. Available data should be used to define the hazards and underlying causes, and relevant safety barriers against these hazards.

3.3. Identify indicators

During the detailed design phase of the AMS, safety indicators should be identified. During the life of the AMS, this step is reiterated, in order to improve the safety monitoring process. New indicators may have to be identified and existing indicators may not be relevant any more or they may have to be adapted to changes in system operation.

The development team identifies three different types of safety indicators in two distinct phases: (i) outcome and (ii) early warning indicators related to safety barriers; and (iii) resilience indicators. Firstly, the indicators based on the safety barriers are established (type i and ii). A review of hazards and planned or implemented safety barriers identifies the most relevant safety barriers. It is not practical to develop safety indicators for all safety barriers. For this purpose, information is obtained from the detailed design phase activities and risk analysis. During risk analysis in the detailed design phase, the need for safety barriers are identified and evaluated, before the required safety barriers are designed in detail. The detailed design documentation of the safety barriers gives input to the process for developing safety indicators. Tests of components and component integration also give input to indicator development by highlighting areas that need special attention in relation to risk.

For each relevant safety barrier, the desired safety goal is described, which summarizes its expected performance and achievements. Outcome indicators reflect a failure of the desired safety outcome, e.g., an accident, near miss, incident. A description of critical elements of the safety barrier gives input to the development of early warning indicators. Early warning indicators reflect the performance of critical elements of the safety barrier, e.g., the performance of associated subsystems. For each relevant safety barrier, at least one outcome and one early warning indicator are required. All proposed indicators should be evaluated against the criteria in Table 1. If none of the proposed indicators fulfills the criteria, the development team has to reiterate steps 3.2 and 3.3 in Fig. 1.

In the second phase of the indicator identification step, resilience indicators (type iii) that complement the early warning indicators related to safety barriers are identified. The resilience indicators are also early warning indicators, but are not related to safety barriers. Hence, they are called resilience indicators in the following. Each of the already identified early warning indicators related to safety barriers (type ii) is associated with one CSF and a corresponding general issue (cf. Section 2). For AUV, the CSF and general issues are adapted from REWI [66]. These are the following: Risk understanding – information about quality of barrier support functions, risk understanding – information about quality of barriers, anticipation – risk/ hazard identification, attention – changes, response – flexibility of organizational structure, robustness – communication between actors, resourcefulness/ rapidity – adequate ICT systems, decisions support – adequate ICT decision support systems, redundancy – redundancy in information processing. In order to represent the planning process of an AUV mission, a new general issue, called mission/ operation characteristics, is added to the CSF anticipation.

The development team assesses suitable indicators for the general issues. Each general issue should be covered by at least one early warning indicator, which means that those general issues not covered by the early warning indicators from phase one should be covered by resilience indicators in phase two. Evaluation of all resilience indicators against the criteria in Table 1 is necessary in order to ensure a usable set of indicators. If not enough resilience indicators satisfy the criteria, the steps 3.4 and 3.5 in Fig. 1 must be reiterated, in order to achieve a comprehensive set of safety early warning indicators.

Each safety indicator has to be thoroughly described. The description should include several aspects: the desired (qualitative) safety goal, critical elements associated with the indicators, data requirements, data sources, sampling intervals, indicator thresholds, safety improvement measures if critical thresholds are reached, and relevant references. Before data and information for the indicators can be collected, necessary interfaces, procedures and processes have to be defined and implemented. This influences the detailed design phase (Fig. 2), because it is necessary to ensure that these interfaces are designed appropriately.

One important aspect of using safety indicators is the sampling interval [27]. Three sampling categories should be considered: short-term, mid-term and long term. Collection of data for short-term indicators occurs at least once per day, but could also be every second or minute. Sampling of data for mid-term indicators occurs at least once a week, but not more often than once per day. Long-term indicators are monitored at least once a month (30 days), but not more often than once a week. Any early warning indicators collected less than once a month might be dismissed from further inclusion in the safety indicator system [66].

Determining the indicator thresholds is another challenge. Hassan and Khan [15], for example, use four classes of risk, which are associated with an index range: Extreme, high, medium, and low. For safety indicators, critical, low, medium, and high, are proposed as classes or thresholds. "Critical", for example, means that the safety threshold is very close to being violated, whereas "high" means that the safety performance is good. Another example for deriving threshold values is given by Saqib and Siddiqi [45], using percentiles of defined requirements. For each safety indicator, such thresholds should be defined individually. Table 2 presents threshold examples for outcome, early warning, and resilience indicators.

3.4. Implement indicators

The implementation of the indicator system has to be prepared in the detailed design phase, in order to provide the right interfaces for collection of data and measurement of the indicators. If the implementation is started too late in the detailed design phase, design reviews might be necessary during construction and commissioning, which may delay the completion of the AMS. Information that is already collected should be used, if possible. Ideally, automated systems should be in place to collect data for short-term safety indicators and evaluate them. Otherwise the indicators may be too resource intensive to be used efficiently and distract the operators and indicator system responsible from their actual tasks.

During AMS operation, the safety indicator system is used and reviewed regularly. Data has to be collected and evaluated on a regular basis. Analysis of the absolute indicator values reflects the safety level during a specific period. Indicators that are measured in a low or critical safety class trigger the defined safety improvement measures

Table 2
Examples of safety indicator thresholds, based on [15] and [45].

Safety rank	Safety class	Safety threshold (exemplary)	
		Early warning indicator or resilience indicator [%]	Outcome indicator [# of occurrences in a period]
1	Critical	0–75	5
2	Low	> 75–85	3
3	Medium	> 85–95	1
4	High	> 95	0

with respect to upper and lower thresholds. These safety improvement measures are dependent on the type of safety indicator, but will lead to input for system improvement.

Trend analysis might add additional information to the monitoring of safety [29]. Especially for indicators, which cannot be measured often, trends might indicate a degradation of the system before thresholds are exceeded. Some outcome indicators represent undesired incidents and do not occur often. Hence, capturing and analyzing data may prove to be difficult. A comparison of outcome and early warning indicators' development gives information on how well the early warning indicators reflect actual safety performance. If their developments differ too much from each other, a review of the set of indicators is necessary.

Øien et al. [66] propose quarterly reporting to follow up on the safety indicators. This is the task of the indicator system responsible. He or she should also present the results to management and initiate discussion of necessary safety improvement measures to be taken in order to improve the safety level. This discussion should involve relevant personnel, e.g., managers, operators, technicians, or engineers. Cause analysis of undesired outcomes can give input to finding more suitable safety indicators [62].

3.5. Review and update

The last step of the process for developing safety indicators is to review the indicators and their implementation regularly during the AMS operation phase. This ensures that the indicators reflect operation and overreliance effects are counteracted [66]. This also requires a review of hazards and operational conditions, i.e., have modifications been undertaken, or new hazards been identified. Input from field tests, the operators and operational data give insights into safety relevant issues that need to be monitored. A workshop approach, as used in the development phase, might add value to the review. Especially, feedback from those gathering data and monitoring the indicators might lead to an improved safety indicator system. Thresholds can be adapted and refined with the operational experience collected. New indicators can be identified and implemented, in order to improve the safety monitoring of the AMS. Discarding and replacement of inadequate and inefficient indicators is one of the tasks. The documentation of the indicator system should reflect how and why changes have been executed. This knowledge is valuable for future indicator systems and enable the organization to build better safety indicator systems for AMS.

4. Exemplification of the process for developing safety indicators

This Section exemplifies the use of the presented process for developing safety indicators based on operation of an AUV, i.e., the REMUS 100, which is discussed, e.g., in [14,47,50]. NTNU operates one REMUS 100 through the AUR Lab [2]. AUVs are used, for example, in mine counter operations, seafloor mapping, medium- and large-scale surveys of seawater properties, and inspection of subsea installations [59]. AUVs are cigar-shaped and follow a pre-programmed mission path. The operators supervise the AUV onshore or onboard a ship or a working vessel. The AUV should detect unexpected or undesired events, abort the mission and return to a meeting point. However, operators might also have to abort the mission, due to deteriorating performance or deteriorating (environmental) conditions. In this case, the operators detect problems and react appropriately. Furthermore, if a mission is finished or aborted, automatically or manually, operators have to be prepared to retrieve the AUV at a meeting point.

Operators carry out maintenance, mission preparation and planning, the missions itself and post mission tasks. Operation here refers to six different phases: mission planning and preparation, deployment, mission execution, retrieval and post mission tasks, inspection and maintenance, and data and mission analysis. Loss of an AUV may occur during deployment, mission execution or retrieval. All phases of an operation are relevant to consider with respect to development of indicators. Currently, measurement and trending of some indicators may have to take place after a mission, since not all data is submitted from the AUV to the operators during a mission.

The application of the process for developing safety indicators is covered only superficially with respect to the organizational arrangements, updating, and review. The focus of this example is on identification of indicators and considerations for implementation.

4.1. Organizational arrangements and scope

The safety indicator system aims at reflecting the safety level of operation of a REMUS 100 AUV. It focuses on the operators and their ability to handle unexpected situations and the recovery of the AUV. Loss of the AUV is the main hazard. Causes for loss can be faults of internal (electronic) components, intrusion of water in the AUV, and wrong planning [47,50]. Immediate causes for internal faults, can be found in setup errors, faulty components, unforeseen interactions and software faults [54]. Causes for water intrusion might be damages due to improper handling, collision, maintenance or through improper sealing of the propulsion system [47,50]. Causes for insufficient planning are typically erroneous estimation of environmental factors, erroneous implementation of parameters and waypoints, and insufficient solving of existing faults [30,47,50,54].

Table 3 gives an overview of hazards for AUV operation and associated safety barriers, adapted from [20]. The Table summarizes the safety barriers in the left column and associated hazards and basic causes in the right columns. Inspection and maintenance refer to the detection and subsequent repair of damages and degradations of the AUV. Procedures refer to the instructions given to the operators, to ensure appropriate maintenance and inspection, correct planning, correct set up of the AUV, and solving of existing faults of the AUV. Instrumentation and alarms refer to self-tests and sensors that detect if the AUV is working as supposed and indicate this to the operator. Communication includes the exchange of safety critical information between the AUV and operators, and among operators. Emergency arrangements refer to those actions that have to be taken after a self-test has detected a critical fault and the retrieval of the AUV after a mission.

4.2. Identify indicators

Table 3
Hazards and safety barriers for AUV operations, adapted from [20] and based on [30,47,50,54].

Safety barriers	**Causes for loss of AUV**		
	Water intrusion	**Insufficient planning**	**Internal faults**
1. Inspection and maintenance	x		x
2. Procedures for:			
Mission preparation	x	x	
Operation of the AUV			x
3. Instrumentation and alarms	x	x	x
4. Communication:			
Between AUV and operators			x
Between operators	x	x	
5. Emergency arrangements	x	x	x

The safety barrier instrumentation and alarms exemplify the

Table 4
Evaluation of proposed safety indicators for the safety barrier instrumentation and alarms.

Safety indicator evaluation criterion	Safety indicators			
	Number of times safety critical faults do not lead AUV to abort mission	**Number of times water detection sensors inside the AUV do not detect water intrusion**	**Percentage of faults related to critical subsystems detected by self-tests**	**Percentage of time critical sensors work without fault**
1	YES, if the AUV is not aborting automatically, it will be difficult for the operator to identify the situation as critical.	YES, if a water ingress in the AUV body is not detected, the AUV is highly endangered.	YES, faults in critical subsystems that are detected are known and can be catered for.	YES, a high availability of sensor systems gives confidence that abnormal situations will be detected.
2	NO, difficult to measure. A proper definition of safety critical alarm is necessary.	YES, if the AUV is retrieved and maintained, the water intrusion will be found. This is a rare event.	NO, not all critical faults might be detected after a mission, without the self-test.	YES, faults of sensors are readily recorded and can be observed.
3	Data can be extracted from fault and mission logs.	Data can be extracted from fault, mission and maintenance logs.	Data can be extracted from fault, mission and maintenance logs.	Data can be extracted from fault and mission logs.
4	PARTLY, data might be subject to interpretation and hence different values may be produced.	PARTLY, mission and maintenance documentation provides unambiguous data.	PARTLY, measurements are subject to evaluation and interpretation of data.	YES.
5	NO, due to the manual evaluation and assessment of faults, the indicator might be subject to different interpretations and manipulation.	PARTLY, if the maintenance logs are not kept properly, the indicator might be manipulated.	NO, might be subject to manipulation, due to detectability of the faults.	YES, data is recorded automatically.
	NOT SELECTED, difficult to implement and measure.	SELECTED, the indicator is specific enough to reflect safety of operation.	NOT SELECTED, measurement difficult and ambiguous.	SELECTED

further steps of the process for developing safety indicators related to safety barriers. The AUV is equipped to detect leaks, ground faults, temperatures and pressures out of operational limits. Ideally, sensors detect faults and trigger alarms that indicate these faults through the monitoring interface to the operator; however, false alarms may occur. Based on Table 3 and the above description, two outcome indicators can be identified: *Number of times water detection sensors inside the AUV do not detect water intrusion* and *number of times safety critical faults do not lead AUV to abort mission*. One critical element of the safety barrier instrumentation and alarms is that the AUV's sensors detect its current state correctly and sufficiently. A second critical element is that alarms are activated in a timely manner and that they raise sufficient awareness of the operator. *Percentage of faults related to critical subsystems detected by self-tests*, and *percentage of time critical sensors work without fault,* are therefore two possible early warning indicators. Table 4 evaluates the four proposed safety indicators, for the safety barrier instrumentation and alarms, against the requirements set in Table 1.

The evaluation in Table 4 shows that a suitable outcome indicator is *number of times water detection sensors inside the AUV do not detect water intrusion*. Sensors in the lower half of the AUV should detect water intrusion, leading to an immediate mission abort when they detect water. If these should not work, the operators would detect water intrusion after the mission during cleaning and inspection of the AUV. A suitable early warning indicator is the *percentage of time critical sensors work without fault*. Examples of critical sensors are leak detection and grounding error detection.

Table 5 describes these two selected safety indicators in detail for use in the safety indicator system. The description contains the required elements stated in Section 3.3. The desired safety goal of the safety barriers describes their expected performance. In respect to the two selected safety indicators, critical sensors should operate during a mission and warn if an undesired event occurs. For both safety indicators, it is critical that the sensors are set up and calibrated to detect undesired events and that they react timely to an undesired event and trigger associated alarms. For the early warning indicator *percentage of time critical sensors work without fault*, it is important to define and select these critical sensors and associated fault messages in the fault logs. The *percentage of time critical sensors work without fault* can be sampled during a mission or after a mission. Since water intrusion is a rare event, *number of times water detection sensors inside the AUV do not detect water intrusion* can only be sampled monthly. If one of the safety indicators should be found in the critical or low safety class, the associated actions described in Table 5 should come into action. In the case of the two selected safety indicators, the causes for the faults should be identified and actions taken against reoccurrence. References for such an investigation might be found in the manuals of the AUV.

All safety barriers should have at least one outcome and one early warning indicator (cf. Section 3.3). Thus, Table 6 and Table 7 propose a set of outcome indicators and early warning indicators for all five types of safety barriers, respectively (cf. Table 3). O3 and EW3 are described in detail. The other identified safety indicators are not detailed here, due to space limits.

The early warning indicators related to safety barriers cover the CSF: risk understanding, robustness and response. Table 8 proposes resilience indicators related to the remaining CSF: Anticipation, attention, resourcefulness/ rapidity, decision support and redundancy. Relevant general issues were selected, based on their suitability for AUV operation. The resilience indicators were developed and refined in order to reflect AUV operation for these general issues. The resilience indicators in Table 8 are motivated by [66], and focus on the adequacy of the ICT system and associated functions, but also organizational learning and awareness for the environment. Fig. 3, adapted from [66], visualizes how the identified early warning indicators presented in Table 7 and the resilience indicators in Table 8 are linked to the CSFs

Table 5
Description of selected safety indicators for the safety barrier instrumentation and alarms.

		O3: Number of times water detection sensors inside the AUV do not detect water intrusion	**EW 3: Percentage of time critical sensors work without fault**
Desired safety goal		If water should enter the sealed AUV body this has to be detected, mission aborted and a warning sent to the operators.	Sensors covering vital functions of the AUV should work continuously during a mission and detect relevant faults if they occur.
Critical elements		Sensors have to react to small amounts of water entering the body. Alarms have to be triggered immediately and a notification send to the operators.	Adequate thresholds for relevant sensors to trigger alarms. Adequate sensors for operating conditions.
Data requirements		–	Definition of critical sensors necessary and identification of associated faults recorded in the fault logs.
Data sources		Water intrusion has to be identified manually and compared with fault logs.	Fault logs and mission logs.
Sampling intervals		Monthly.	During or after mission.
Thresholds	Critical	2 and more are critical	≤97.5
	Low	1	> 97.5 – 99.0
	Medium	–	> 99.0 – 99.5
	High	0	≥99.5
Actions		Identify causes for water intrusion. Implement measures against reoccurrence. Send in AUV to supplier for repair.	Identify main contributors to the decreased performance. Identify causes and implement measures against reoccurrence.
References		Manuals for maintenance and inspection	Manuals for maintenance, inspection and operation
Associated resilience attribute, CSF and general issue		None – outcome indicator	Risk awareness – risk understanding – information about the quality of barriers

Table 6
Proposed outcome indicators for all identified safety barriers of AUV operation.

Outcome indicator		**Safety barriers**	**Sampling interval**
O1	Number of faults that can be traced back to erroneous or lacking maintenance	Inspection and maintenance	Monthly
O2	Number of incidents where necessary procedures were not available during a mission	Procedures	Monthly
O3	Number of times water detection sensors inside the AUV do not detect water intrusion	Instrumentation and alarms	Monthly
O4	Percentage of missions where connection between operators and AUV was lost unplanned for more than 30 min	Communication	Monthly
O5	Number of (temporary) losses of AUV	Emergency procedures	Monthly

Table 7
Proposed early warning indicators for all identified safety barriers of AUV operation.

Early warning indicator		**Safety barriers**	**Resilience attribute – CSF – general issue**	**Sampling interval**
EW1	Percentage of maintenance and inspections completed in specified periods	Inspection and maintenance	Risk awareness – risk understanding – information about quality of barrier support functions	Monthly
EW2	Percentage of procedures updated and revised in the designated periods	Procedures	Risk awareness – risk understanding – information about quality of barrier support functions	Monthly
EW3	Percentage of time, critical sensors work without fault	Instrumentation and alarms	Risk awareness – risk understanding – information about quality of barriers	During or after a mission
EW4	Percentage of anticipated status messages received from the AUV	Communication	Response capacity – robustness – communication between actors	During or after a mission
EW5	Percentage of successful recoveries of AUV within 15 min after end of mission or preliminary mission abort	Emergencyprocedures	Response capacity – response – flexibility of organizational structure	Monthly

and general issues.

4.3. Indicator implementation

Nine missions of the NTNU AUR Lab were analyzed for gathering input data for testing the indicators. The data were recorded in the electronic mission and fault logs, which are created by the AUV. The NTNU AUR Lab carried out these missions between 06. August and 19. November 2015. The analysis revealed which information is already recorded in the electronic mission or fault logs and which information might be recorded or extracted with some additional effort. Table 9 summarizes the data availability.

Several safety indicators can be captured automatically from the electronic mission and fault logs, e.g., O3, EW1, EW3, EW4, EW5, R2, R3 and R5. Algorithms for their automatic evaluation would reduce the manual work associated with the safety indicator system. Several of the proposed safety indicators need manual collection, e.g., from an AUV journal or a computerized maintenance management system, where operators record performed inspections/ maintenance (for EW1), incidents before or during operation (O1, R4), and changes in procedures (EW2). Other safety indicators can also benefit from such documentation, especially the outcome indicators. Procedures and programs for collection of data for the indicators still need to be implemented for several of the proposed safety indicators. Hence, not all safety indicators could be assessed for the NTNU AUR Lab missions.

Fig. 4 presents the number of recorded faults per hour of operation for each of the missions. None of these faults is relevant for the sensor system. Most of the recorded faults correspond to warnings, e.g., problems in the compass bias table, or the “vehicle stuck on surface; attempting to drive it down”. These fault messages are common warnings, and do not affect the mission execution, because the AUV is not endangered, c.f. [14]. Only mission number 6 had to be aborted, due to a failure in the thrusters. Causes and subsequent actions were not recorded, which means that causal analysis is not possible. Hence, the current documentation would need to be adapted to use the proposed indicators efficiently.

The indicator system responsible should carry out evaluation of the safety indicators and prepare the reports and distribution of the results. If trends or safety indicator values show degradation of operation, safety improvement measures have to be taken to improve operation. Additionally, incidents and problems should be discussed with relevant stakeholders. For example, for the indicator R2, two relevant fault messages are recorded. These are “Vehicle at low altitude. Executing emergency climb”, and “Vehicle stuck on bottom, attempting to float free”. Several instances of these have been recorded in the missions 4, 5, and 8, shown as “contacts with seabed” in Fig. 5. This shows that in three missions assessment of the environment might have been insufficient. Especially mission number 5 had a high rate of contacts between AUV and Seabed. For that mission, it should be analyzed why so many contacts occurred and how that could be prevented in the future in the planning process of a mission. EW 4 can be directly assessed from the fault logs. During the nine recorded missions, no critical sensors failed. Hence, the safety indicator did not reveal any deficiencies. During the next review, this early warning indicator should be checked for relevance, since critical sensor faults seem not to occur often enough to indicate safe operation.

Table 8
Proposed resilience indicators for AUV operation, motivated by [43].

	Resilience indicator	Resilience attribute – CSF – general issue	Reasoning	Sampling frequency
R1	Percentage of missions that have been discussed in terms of hazards and risks before mission start	Risk awareness – anticipation – risk/ hazard identification	Being aware of possible hazards and risks for a certain area prepares the operators to plan and prepare for the mission accordingly.	Monthly
R2	Number of contacts between AUV and seafloor per hour, during a mission	Risk awareness – anticipation – mission characteristics	Knowing the conditions and characteristics of the mission environment is important in order to set up the AUV correctly for a mission. If the AUV has frequent contact with the sea floor, it was not set up correctly for the topography of the sea floor. This indicates an insufficient planning process.	After a mission
R3	Percentage of missions where environmental conditions exceeded the allowable limits	Risk awareness – attention – changes (environmental)	Monitoring changes is an important task of the operators, especially in respect to sea state and weather.	Monthly
R4	Average time between status messages	Response capacity – resourcefulness/ rapidity – adequate ICT systems	Without adequate knowledge of occurrences, a timely response is not possible.	During or after a mission
R5	Percentage of missions where monitoring laptop was (partly) not available during a mission (e.g., due to low battery)	Support – decision support – adequate ICT decision support	A monitoring laptop displays all critical information about the AUV and allows for change and adaption of the mission plan.	Monthly
R6	Number of alternatively available communication channels between AUV and operators during a mission	Support – redundancy – redundancy in information processing	Without information from the AUV, the operators do not know about the state and intentions of the AUV. Especially during retrieval, where the position must be communicated.	During or after a mission

5. Discussion and conclusion

The proposed process for developing safety indicators in this article is based on two methods from high-risk industries, which are synthesized and adjusted to the application area of AMS. Currently, no structured process for developing safety indicators for AMS exists or is in use. HSE and REWI processes are complementary [67], and the article shows how the two methods can be integrated, adapted to AMS, and applied jointly. The presented process for developing safety indicators focuses efforts, resources and attention to identify a sufficiently comprehensive, but still a manageable set of safety indicators. The dual assurance and REWI methods, if applied separately, would overlook important safety aspects [29,65]. Thus, the proposed process for developing safety indicators finds a coherent set of safety indicators that covers the company, aiming for complete coverage of safety aspects.

This article locates the steps of the process for developing safety indicators in the system life cycle of an AMS. The process for developing safety indicators is most efficient if it is implemented during the design of the AMS, and then further refined based on operational experience. Necessary interfaces and systems for indicator collection can be developed in the detailed design phase, which may reduce implementation costs and benefit the overall system design. The case study shows that implementing the process for developing safety indicators during the operation phase of a system is challenging concerning collection of the safety indicators. Additional effort is necessary to create necessary interfaces, and implement procedures and processes for safety indicator development.

The development team could cooperate with the system safety analysts to establish a relationship between risk assessments and the safety indicators. This would in return overcome some deficiencies of the two methods, as mentioned by Øien [68], for example, the missing link to risk models. Comparison of outcome indicators and early warning indicators helps to evaluate and validate safety performance and to reveal deficiencies in the safety indicator system. If the performance of early warning and outcome indicators differs too much, the safety indicators have to be reviewed with respect to usefulness and efficiency. Generally, the safety indicator system should be reevaluated regularly, in order to improve the system.

In the example of an AUV, the process for developing safety indicators results in five outcome indicators and eleven early warning indicators. Twenty safety indicators is the suggested upper limit by Øien et al. [66] for the REWI method. Likely, there will be more than

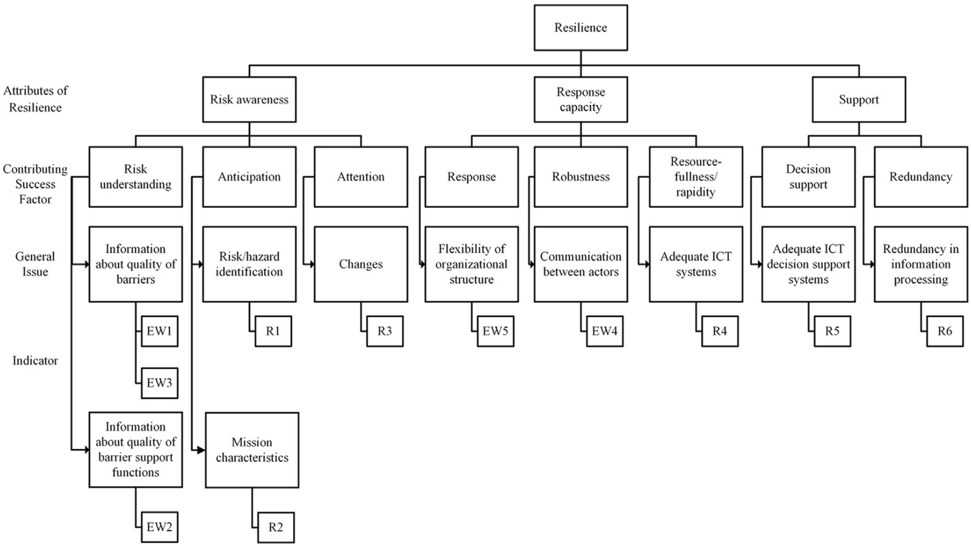

Resilience
Attributes of Resilience
Risk awareness
Response capacity
Support
Contributing Success Factor
Risk understanding
Anticipation
Attention
Response
Robustness
Resource-fullness/ rapidity
Decision support
Redundancy
General Issue
Information about quality of barriers
Risk/hazard identification
Changes
Flexibility of organizational structure
Communication between actors
Adequate ICT systems
Adequate ICT decision support systems
Redundancy in information processing
Indicator
EW1
R1
R3
EW5
EW4
R4
R5
R6
EW3
Information about quality of barrier support functions
Mission characteristics
EW2
R2

Fig. 3. The proposed early warning indicators and resilience indicators related to the resilience attributes, the CSFs, and the general issues, adapted from [66].

Table 9
Data sources for the proposed safety indicators.

Safety Indicators	Data source			
	Already found in mission logs collected by the AUV	Already found in fault logs collected by the AUV	Data collection in the AUV's mission/ fault logs possible	Manual documentation/ collection necessary
O1		Partly		Yes
O2				Yes
O3		Partly		Yes
O4			Partly	Yes
O5	Partly			Yes
EW1				Yes
EW2				Yes
EW3		Yes		
EW4	Partly		Yes	
EW5	Partly		Yes	
R1				Yes
R2		Yes		
R3				Yes
R4	Partly		Yes	
R5				Yes
R6	Partly	Partly	Yes	

20 safety indicators for more complex systems with the suggested process for developing safety indicators. However, if the safety indicators can be collected by a computer system, with little human labor required, more than 20 safety indicators should be manageable. Generally, the amount of safety indicators depends on the target organizational level and the organizational capabilities. The safety indicators in this article cover both direct safety functions, e.g., alarms, and broader aspects of safety functions, such as maintenance, which has an essential influence on safety, even though maintenance alone does not guarantee safe operation [56]. A relationship between safety and the safety indicators is inferred, but not demonstrated. It is assumed that the relationships between the safety indicators and safety in other industries are also valid for operation of AMS.

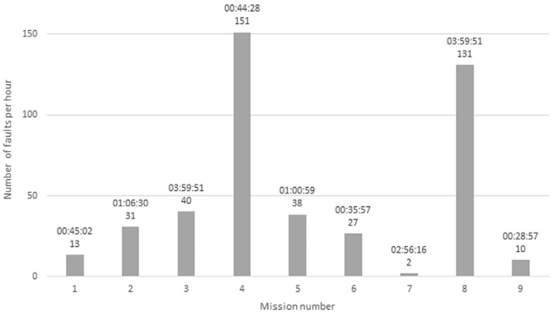


Fig. 4. Number of faults per hour of operation recorded during nine missions of the REMUS 100 of the NTNU AUR Lab between 06. August 2015 and 19. November 2015. Total mission time is displayed above the number of faults.

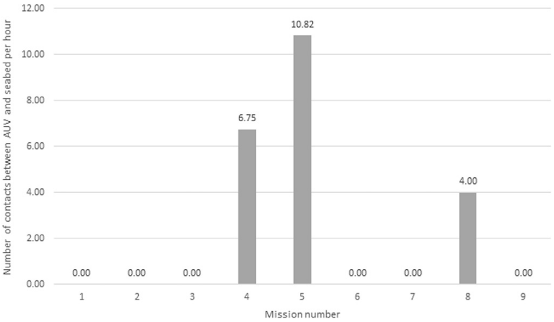


Fig. 5. Number of contacts between AUV and seabed per hour of operation, recorded during nine missions of the REMUS 100 of the NTNU AUR Lab between 06. August 2015 and 19. November 2015.

Regarding the safety indicator development example, some more limitations have to be mentioned. The system was chosen for its simplicity and accessibility as an AMS. The suggested process for developing safety indicators and management of safety indicators may be resource demanding for an organization operating one REMUS 100 AUV, only. Some of the identified safety indicators, however, apply to other AMS, as well. Some safety indicators are similar to the findings of Rødseth et al. [43, p. 30 ff.]. To investigate its capabilities in a broader sense, the proposed process for developing safety indicators should be applied to other AMS, such as autonomous or unmanned ships, or operation of multiple AMS. This can complement efforts, such as Rødseth et al.'s [43], in a structured manner.

Due to changes of season, sea state and weather, it may be difficult to collect some safety indicators regularly and unbiased. Examples are *percentage of missions that have been discussed in terms of hazards and risks before mission start*, *percentage of missions where environmental conditions exceeded the allowable limits,* e.g., *wave height, wind speed*, or *percentage of maintenance and inspections completed in specified periods*. These safety indicators are highly dependent on the amount of missions executed. For AMS, which are operated frequently, such concerns are less relevant.

Most of the proposed safety indicators can be collected from the fault logs, or captured if some more data is recorded automatically. Currently, manual evaluation and investigation is necessary for several safety indicators. This makes the implementation difficult and additional procedures and systems need to be put into operation for the collection of these safety indicators. This applies to, e.g., *number of faults that can be traced back to erroneous or lacking maintenance, percentage of missions that have been discussed in terms of hazards and risks before mission start*, or *number of alternatively available communication channels between AUV and operators during a mission*.

Some of the proposed safety indicators for AUV operation can be sampled in short-term intervals, e.g., *number of alternatively available communication channels between AUV and operators during a mission*, *number of contacts between AUV and seafloor per hour, during a mission,* or *percentage of anticipated status messages received from the AUV*. These safety indicators could be used during operation to assess how well the AMS performs in real-time with respect to safety. Further investigation is necessary to develop and implement a real-time or online safety monitoring systems for AMS. On the other hand, for some of the proposed safety indicators that are not updated often enough, e.g., *percentage of procedures updated and revised in the designated periods*, safety audits might be a more suitable tool. Further investigation is needed regarding the feasibility of both an online safety monitoring system and safety audits for AMS.

Acknowledgements

The authors wish to thank the anonymous reviewers for their valuable comments. The support by the NTNU AUR Lab, providing mission and fault data, is acknowledged. Thanks to Jeevith Hegde for useful comments on a draft version of this article. Comments from the participants of the doctoral degree course *System Safety Engineering and Management* on an early version of this article are appreciated.

This work was supported by the Research Council of Norway through the Centres of Excellence funding scheme, Project number 223254 – AMOS.

References

[1] API . API RP 754 process safety performance indicators for the refining and petrochemical industries, Second edition. Washington D.C: American Petroleum Institute Publishing; 2016.

[2] AUR Lab . The Applied Underwater Robotics Laboratory. ⟨http://www.ntnu.edu/aur-lab⟩. Accessed: 02.02; 2015.

[3] Bainbridge L. Ironies of automation. Automatica 1983;19:775–9. http://dx.doi.org/10.1016/0005-1098(83)90046-8.

[4] Blanchard BS. System engineering management4th ed.. Hoboken, NJ: Wiley; 2008, [9780470167359].

[5] Brito M, Griffiths G, Ferguson J, Hopkin D, Mills R, Pederson R, et al. A behavioral probabilistic risk assessment framework for managing autonomous underwater vehiclevehicle deployments. J Atmos Ocean Technol 2012;29:1689–703. http://dx.doi.org/10.1175/Jtech-D-12-00005.1.

[6] Brito M, Griffiths G. A Bayesian approach for predicting risk of autonomous underwater vehicle loss during their missions. Reliab Eng Syst Safe 2016;146:55–67. http://dx.doi.org/10.1016/j.ress.2015.10.004.

[7] Brito MP, Griffiths GResults of expert judgments on the faults and risks with Autosub3 and an analysis of its campaign to Pine Island Bay, Antarctica, 2009. Proceedings of the International Symposium on Unmanned Untethered Submersible Technology (UUST 2009), Durham, New Hampshire, 23-26 August 2009: Autonomous Undersea Systems Institute (AUSI); 2009. p. [14p]

[8] Brito MP, Griffiths G, Challenor P. Risk analysis for autonomous underwater vehicle operations in extreme environments. Risk Anal: Off Publ Soc Risk Anal 2010;30:1771–88. http://dx.doi.org/10.1111/j.1539-6924.2010.01476.x.

[9] Delatour G, Laclemence P, Calcei D, Mazri C. Safety performance indicators: a Questioning diversity. Chem Eng Trans 2014;36:55–60. http://dx.doi.org/10.3303/Cet1436010.

[10] DHSG . Final report on the investigation of the macondo well blowout. Berkeley, California, USA: Deepwater Horizon Study Group, Center for Catastrophic Risk Management (CCRM), University of California; 2011.

[11] Dyreborg J. The causal relation between lead and lag indicators. Saf Sci 2009;47:474–5. http://dx.doi.org/10.1016/j.ssci.2008.07.015.

[12] Endsley MR. Toward A theory of situationsituation awareness in dynamic-systems. Hum Factors 1995;37:32–64. http://dx.doi.org/10.1518/001872095779049543.

[13] Griffiths G, Brito M. Predicting risk in missions under sea ice with Autonomous Underwater Vehicles. Autonomous Underwater Vehicles, 2008 AUVIEEE/OES2008. p.1-7; 2008.

[14] Griffiths G, Brito M, Robbins I, Moline M. Reliability of two REMUS-100 AUVs based on fault log analysis and elicited expert judgment. Proceedings of the International Symposium on Unmanned Untethered Submersible Technology (UUST 2009), Durham, New Hampshire, 23-26 August. Durham NH, USA: Autonomous Undersea Systems Institute (AUSI); 2009. p. [12p]; 2009.

[15] Hassan J, Khan F. Risk-based asset integrity indicators. J Loss Prev Process Ind 2012;25:544–54. http://dx.doi.org/10.1016/j.jlp.2011.12.011.

[16] Hegde J, Utne IB, Schjølberg I. Applicability of Current Remotely Operated Vehicle Standards and Guidelines to Autonomous Subsea IMR Operations:V007T06A26. http://dx.doi.org/10.1115/omae2015-41620; 2015.

[17] Ho G, Pavlovic N, Arrabito R. Human Factors Issues with Operating Unmanned Underwater Vehicles. Proceedings of the Human Factors and Ergonomics Society Annual Meeting;55:429-433.http://dx.doi.org/10.1177/1071181311551088; 2011.

[18] Hollnagel E. Human reliability assessment in context. Nucl Eng Technol 2005;37:159.

[19] Hopkins A. Thinking about process safety indicators. Saf Sci 2009;47:460–5. http://dx.doi.org/10.1016/j.ssci.2007.12.006.

[20] HSE , CIA . Developing process safety indicators: a step-by-step guide for chemical and major hazard industries. 1. Norwich: Chemical Industries Association (CIA) and Health and Safety Executive (HSE); 2006.

[21] Huang H-M, Messina E, Jacoff A, Wade R, McNair M. Performance measures framework for unmanned systems (PerMFUS): models for contextual metrics. In: Proceedings of the 10th Performance Metrics for Intelligent Systems Workshop. Baltimore, Maryland: ACM. p. 22-28; 2010.

[22] IAEA . Operational safety performance indicators for nuclear power plants IAEA-TECDOC. Vienna: International Atomic Energy Agency; 2000. p. 75.

[23] Insaurralde CC, Lane DM. Autonomy-assessment criteria for underwater vehicles. Auton Underw Veh (AUV) 2012:1–8.

[24] ISO ISO. 31000 risk management - principles and guidelines. Int Stand Organ 2009.

[25] Kjellen U. Prevention of accidents through experience feedback. Hoboken: Hoboken: Taylor and Francis; 2000, [9780748409259].

[26] Kjellen U. The safety measurement problem revisited. Saf Sci 2009;47:486–9. http://dx.doi.org/10.1016/j.ssci.2008.07.023.

[27] Knegtering B, Pasman H. The safety barometer. J Loss Prev Process Ind 2013;26:821–9. http://dx.doi.org/10.1016/j.jlp.2013.02.012.

[28] Leveson N. A systems approach to risk management through leading safety indicators. Reliab Eng Syst Safe 2015;136:17–34. http://dx.doi.org/10.1016/j.ress.2014.10.008.

[29] Leveson NG. Engineering a safer world - system thinking applied to safety. Cambridge, Massachusetts, USA; London, England: The MIT Press; 2011.

[30] Manley JE. The Role of Risk in AUV Development and Deployment. OCEANS - Europe2007. p. 1-6; 2007.

[31] MUNIN. Maritime Unmanned Navigation through Intelligence in Networks. ⟨http://www.unmanned-ship.org/munin/⟩. Accessed: 23.07. 2015; 2012.

[32] Nilssen I, Odegard O, Sorensen AJ, Johnsen G, Moline MA, Berge J. Integrated environmental mapping and monitoring, a methodological approach to optimise knowledge gathering and sampling strategy. Mar Pollut Bull 2015;96:374–83. http://dx.doi.org/10.1016/j.marpolbul.2015.04.045.

[33] Norgren P, Lubbad R, Skjetne R. Unmanned underwater vehicles in Arctic operations. In: Proceedings of the 22nd IAHR International Symposium on Ice. Singapore. p. 89–101; 2014.

[34] OECD. Guidance On Developing Safety Performance Indicators related to Chemical Accident Prevention, Preparedness and Response, for Industry. In: Environment Directorate OFECAD, editor. Series on Chemical Accidents. second edition ed. Paris: Environment Directorate, Organisation For Economic Cooperation And Development; 2008. p. 156.

[35] OGP . Process safety - recommended practice on key performance indicators. London, Brussels: International Association of Oil and Gas Producers; 2011. p. 36.

[36] Paltrinieri N, Oien K, Cozzani V. Assessment and comparison of two early warning indicator methods in the perspective of prevention of atypical accident scenarios. Reliab Eng Syst Safe 2012;108:21–31. http://dx.doi.org/10.1016/

j.ress.2012.06.017.

[37] Porathe T. Remote Monitoring and Control of Unmanned Vessels – The MUNIN Shore Control Centre. In: Bertram V, editor. 13th International Conference on Computer and IT Applications in the Maritime Industries. Redworth: Technische Universität Hamburg-Harburg; 2014. p. 460–7.

[38] Porathe T, Prison J, Man Y. Situation awareness in remote control centres for unmanned ships. human factors in ship design & operation. London, UK: The Royal Institution of Naval Architects; 2014.

[39] Rausand M. Risk assessment - theory, methods, and applications, 1 ed.. Hoboken, New Jersey, USA: John Wiley & Sons; 2011, [978-0-470-63764-7].

[40] Reason JT. Managing the risks of organizational accidents. Farnham: Surrey Ashgate Aldershot; 1997, [978 1 84014 105 4].

[41] Reiman T, Pietikainen E. Leading indicators of system safety - Monitoring and driving the organizational safety potential. Saf Sci 2012;50:1993–2000. http://dx.doi.org/10.1016/j.ssci.2011.07.015.

[42] Rosness R, Grøtan TO, Guttormsen G, Herrera IA, Steiro T, Størseth F, et al. Organisational accidents and resilient organisations: six perspectives revision 2, 2 ed.. Trondheim: SINTEF Technology and Society; 2010. p. 141.

[43] Rødseth ØJ, Tjora Å, Baltzersen P. D4.5 Architecture specification. Maritime Unmanned Navigation throughIntelligence in Networks; 2014.

[44] Rødseth ØJ, Burmeister H-C. Risk assessment for an unmanned merchant ship. TransNav, Int J Mar Navig Saf Sea Transp 2015;9:357–64. http://dx.doi.org/10.12716/1001.09.03.08.

[45] Saqib N, Siddiqi MT. Thresholds and goals for safety performance indicators for nuclear power plants. Reliab Eng Syst Safe 2005;87:275–86. http://dx.doi.org/10.1016/j.ress.2004.05.006.

[46] Skogdalen JE, Utne IB, Vinnem JE. Developing safety indicators for preventing offshore oil and gas deepwater drilling blowouts. Saf Sci 2011;49:1187–99. http://dx.doi.org/10.1016/j.ssci.2011.03.012.

[47] Stokey R, Austin T, von Alt C, Purcell M, Goldsborough R, Forrester N. et al.AUV Bloopers or Why Murphy Must have been an Optimist: A Practical Look at Achieving Mission Level Reliability in an Autonomous Underwater Vehicle. Proceedings of the International Symposium on Unmanned Untethered Submersible Technology.New Hampshire; 1999.

[48] Swuste P, Theunissen J, Schmitz P, Reniers G, Blokland P. Process safety indicators, a review of literature. J Loss Prev Process Ind 2016;40:162–73. http://dx.doi.org/10.1016/j.jlp.2015.12.020.

[49] Thieme CA, Utne IB, Schjølberg I. Risk modeling of autonomous underwater vehicle operation focusing on the human operator. In: Podofillini L, Sudret B, Stojadinovic B, Zio E, Kröger W, editors. 25th European Safety and Reliability Conference, ESREL. Zürich, Switzerland: CRC Press, Taylor & Francis Group; 2015. p. 3653–60.

[50] Thieme CA, Utne IB, Schjølberg I A Risk Management Framework For Unmanned Underwater Vehicles Focusing On Human And Organizational Factors Proceedings of the ASME 2015 34th International Conference on Ocean, Offshore and Arctic Engineering OMAE2015. St. John's, NL, Canada: ASME; 2015

[51] Tinmannsvik RK, Øien K, Størseth F. Building safety by resilient organization - A case specific approach. In: Bris R, Guedes Soares C, Martorell S, editors. Reliability, Risk, and Safety, Three Volume Set. Prague: CRC Press; 2009. p. 1209–14.

[52] Navy US. The Navy Unmanned Undersea Vehicle (UUV) Master Plan. United States of America Department of the Navy; 2004.

[53] US Navy. The Navy Unmanned Surface Vehicle (USV) Master Plan. 1 ed2007.

[54] Utne IB, Schjølberg I. A Systematic Approach To Risk Assessment - Focusing On Autonomous Underwater Vehicles And Operations In Arctic Areas. ASME In: Proceedings of the 33rd International Conference on Ocean, Offshore and Arctic Engineering. San Francisco, California, USA2014; 2014.

[55] Vagia M, Transeth AA, Fjerdingen SA. A literature review on the levels of automation during the years. What are the different taxonomies that have been proposed?. Appl Ergon 2016;53(Part A):190–202. http://dx.doi.org/10.1016/j.apergo.2015.09.013.

[56] Vinnem JE. Risk indicators for major hazards on offshore installations. Saf Sci 2010;48:770–87. http://dx.doi.org/10.1016/j.ssci.2010.02.015.

[57] Vinnem JE, Utne IB, Schjolberg I. On the need for online decision support in FPSO-shuttle tanker collision risk reduction. Ocean Eng 2015;101:109–17. http://dx.doi.org/10.1016/j.oceaneng.2015.04.008.

[58] Woods DD. Essential characteristics of Resilience. In: Hollnagel E, Woods DD, Leveseon NG, editors. Engineering -Concepts and Precepts. 1. Surrey, UK; Burlington, USA: Ashgate; 2006. p. 21–34, [987-0-7546-4904-5].

[59] Yuh J, Marani G, Blidberg DR. Applications of marine robotic vehicles. Intell Serv Robot 2011;4:221–31. http://dx.doi.org/10.1007/s11370-011-0096-5.

[60] Øien K. A framework for the establishment of organizational risk indicators. Reliab Eng Syst Safe 2001;74:147–67. http://dx.doi.org/10.1016/S0951-8320(01)00068-0.

[61] Øien K. Risk indicators as a tool for risk control. Reliab Eng Syst Safe 2001;74:129–45. http://dx.doi.org/10.1016/S0951-8320(01)00067-9.

[62] Øien K. Development of early warning indicators based on incident investigation. In: Proceedings of the 9th International Conference on Probabilistic Safety Assessment and Management, PSAM 20082008. p. 1809–1816; 2008.

[63] Øien K, Massaiu S, Tinmannsvik RK, Størseth F. Development of early warning indicators based on Resilience Engineering. In: Proceedings of the 10th International Conference on Probabilistic Safety Assessment and Management, PSAM 20102010. p. 1762–1771; 2010.

[64] Øien K, Utne IB, Herrera IA. Building Safety indicators: Part 1 - Theoretical foundation. Saf Sci 2011;49:148–61. http://dx.doi.org/10.1016/j.ssci.2010.05.012.

[65] Øien K, Utne IB, Tinmannsvik RK, Massaiu S. Building Safety indicators: Part 2 - Application, practices and results. Saf Sci 2011;49:162–71. http://dx.doi.org/10.1016/j.ssci.2010.05.015.

[66] Øien K, Massaiu S, Tinmannsvik RK. Guideline for implementing the REWI method. 1.3 ed. Trondheim: SINTEF, IFE; 2012. p. 40.

[67] Øien K, Paltrinieri N. Resilience based indicators - ability to 'cope with the unexpected' Resilience based Early Warning Indicators - complementary to other methods. 1.1 ed: SINTEFTechnologyand Society; 2012.

[68] Øien K. Remote operation in environmentally sensitive areas: development of early warning indicators. J Risk Res 2013;16:323–36. http://dx.doi.org/10.1080/13669877.2012.729523.

Article 6

Thieme, C. A., Utne, I. B. & Schjølberg, I. 2015. *A risk management framework for unmanned underwater vehicles focusing on human and organizational factors.* Proceedings of the ASME 2015 34th International Conference on Ocean, Offshore and Arctic Engineering OMAE2015, 31.05.-05.06.2015. St. John's, NL, Canada. ASME.

This page is intentionally left blank

Part III – Previous PhD Theses Published at the Department of Marine Technology

This page is intentionally left blank

Previous PhD theses published at the Department of Marine Technology

(earlier: Faculty of Marine Technology)

NORWEGIAN UNIVERSITY OF SCIENCE AND TECHNOLOGY

Report No.	Author	Title
	Kavlie, Dag	Optimization of Plane Elastic Grillages, 1967
	Hansen, Hans R.	Man-Machine Communication and Data-Storage Methods in Ship Structural Design, 1971
	Gisvold, Kaare M.	A Method for non-linear mixed -integer programming and its Application to Design Problems, 1971
	Lund, Sverre	Tanker Frame Optimalization by means of SUMT-Transformation and Behaviour Models, 1971
	Vinje, Tor	On Vibration of Spherical Shells Interacting with Fluid, 1972
	Lorentz, Jan D.	Tank Arrangement for Crude Oil Carriers in Accordance with the new Anti-Pollution Regulations, 1975
	Carlsen, Carl A.	Computer-Aided Design of Tanker Structures, 1975
	Larsen, Carl M.	Static and Dynamic Analysis of Offshore Pipelines during Installation, 1976
UR-79-01	Brigt Hatlestad, MK	The finite element method used in a fatigue evaluation of fixed offshore platforms. (Dr.Ing. Thesis)
UR-79-02	Erik Pettersen, MK	Analysis and design of cellular structures. (Dr.Ing. Thesis)
UR-79-03	Sverre Valsgård, MK	Finite difference and finite element methods applied to nonlinear analysis of plated structures. (Dr.Ing. Thesis)
UR-79-04	Nils T. Nordsve, MK	Finite element collapse analysis of structural members considering imperfections and stresses due to fabrication. (Dr.Ing. Thesis)
UR-79-05	Ivar J. Fylling, MK	Analysis of towline forces in ocean towing systems. (Dr.Ing. Thesis)
UR-80-06	Nils Sandsmark, MM	Analysis of Stationary and Transient Heat Conduction by the Use of the Finite Element Method. (Dr.Ing. Thesis)

UR-80-09	Sverre Haver, MK	Analysis of uncertainties related to the stochastic modeling of ocean waves. (Dr.Ing. Thesis)
UR-81-15	Odland, Jonas	On the Strength of welded Ring stiffened cylindrical Shells primarily subjected to axial Compression
UR-82-17	Engesvik, Knut	Analysis of Uncertainties in the fatigue Capacity of Welded Joints
UR-82-18	Rye, Henrik	Ocean wave groups
UR-83-30	Eide, Oddvar Inge	On Cumulative Fatigue Damage in Steel Welded Joints
UR-83-33	Mo, Olav	Stochastic Time Domain Analysis of Slender Offshore Structures
UR-83-34	Amdahl, Jørgen	Energy absorption in Ship-platform impacts
UR-84-37	Mørch, Morten	Motions and mooring forces of semi submersibles as determined by full-scale measurements and theoretical analysis
UR-84-38	Soares, C. Guedes	Probabilistic models for load effects in ship structures
UR-84-39	Aarsnes, Jan V.	Current forces on ships
UR-84-40	Czujko, Jerzy	Collapse Analysis of Plates subjected to Biaxial Compression and Lateral Load
UR-85-46	Alf G. Engseth, MK	Finite element collapse analysis of tubular steel offshore structures. (Dr.Ing. Thesis)
UR-86-47	Dengody Sheshappa, MP	A Computer Design Model for Optimizing Fishing Vessel Designs Based on Techno-Economic Analysis. (Dr.Ing. Thesis)
UR-86-48	Vidar Aanesland, MH	A Theoretical and Numerical Study of Ship Wave Resistance. (Dr.Ing. Thesis)
UR-86-49	Heinz-Joachim Wessel, MK	Fracture Mechanics Analysis of Crack Growth in Plate Girders. (Dr.Ing. Thesis)
UR-86-50	Jon Taby, MK	Ultimate and Post-ultimate Strength of Dented Tubular Members. (Dr.Ing. Thesis)
UR-86-51	Walter Lian, MH	A Numerical Study of Two-Dimensional Separated Flow Past Bluff Bodies at Moderate KC-Numbers. (Dr.Ing. Thesis)
UR-86-52	Bjørn Sortland, MH	Force Measurements in Oscillating Flow on Ship Sections and Circular Cylinders in a U-Tube Water Tank. (Dr.Ing. Thesis)
UR-86-53	Kurt Strand, MM	A System Dynamic Approach to One-dimensional Fluid Flow. (Dr.Ing. Thesis)
UR-86-54	Arne Edvin Løken, MH	Three Dimensional Second Order Hydrodynamic Effects on Ocean Structures in Waves. (Dr.Ing. Thesis)

UR-86-55	Sigurd Falch, MH	A Numerical Study of Slamming of Two-Dimensional Bodies. (Dr.Ing. Thesis)
UR-87-56	Arne Braathen, MH	Application of a Vortex Tracking Method to the Prediction of Roll Damping of a Two-Dimension Floating Body. (Dr.Ing. Thesis)
UR-87-57	Bernt Leira, MK	Gaussian Vector Processes for Reliability Analysis involving Wave-Induced Load Effects. (Dr.Ing. Thesis)
UR-87-58	Magnus Småvik, MM	Thermal Load and Process Characteristics in a Two-Stroke Diesel Engine with Thermal Barriers (in Norwegian). (Dr.Ing. Thesis)
MTA-88-59	Bernt Arild Bremdal, MP	An Investigation of Marine Installation Processes – A Knowledge - Based Planning Approach. (Dr.Ing. Thesis)
MTA-88-60	Xu Jun, MK	Non-linear Dynamic Analysis of Space-framed Offshore Structures. (Dr.Ing. Thesis)
MTA-89-61	Gang Miao, MH	Hydrodynamic Forces and Dynamic Responses of Circular Cylinders in Wave Zones. (Dr.Ing. Thesis)
MTA-89-62	Martin Greenhow, MH	Linear and Non-Linear Studies of Waves and Floating Bodies. Part I and Part II. (Dr.Techn. Thesis)
MTA-89-63	Chang Li, MH	Force Coefficients of Spheres and Cubes in Oscillatory Flow with and without Current. (Dr.Ing. Thesis
MTA-89-64	Hu Ying, MP	A Study of Marketing and Design in Development of Marine Transport Systems. (Dr.Ing. Thesis)
MTA-89-65	Arild Jæger, MH	Seakeeping, Dynamic Stability and Performance of a Wedge Shaped Planing Hull. (Dr.Ing. Thesis)
MTA-89-66	Chan Siu Hung, MM	The dynamic characteristics of tilting-pad bearings
MTA-89-67	Kim Wikstrøm, MP	Analysis av projekteringen for ett offshore projekt. (Licenciat-avhandling)
MTA-89-68	Jiao Guoyang, MK	Reliability Analysis of Crack Growth under Random Loading, considering Model Updating. (Dr.Ing. Thesis)
MTA-89-69	Arnt Olufsen, MK	Uncertainty and Reliability Analysis of Fixed Offshore Structures. (Dr.Ing. Thesis)
MTA-89-70	Wu Yu-Lin, MR	System Reliability Analyses of Offshore Structures using improved Truss and Beam Models. (Dr.Ing. Thesis)
MTA-90-71	Jan Roger Hoff, MH	Three-dimensional Green function of a vessel with forward speed in waves. (Dr.Ing. Thesis)
MTA-90-72	Rong Zhao, MH	Slow-Drift Motions of a Moored Two-Dimensional Body in Irregular Waves. (Dr.Ing. Thesis)

MTA-90-73	Atle Minsaas, MP	Economical Risk Analysis. (Dr.Ing. Thesis)
MTA-90-74	Knut-Aril Farnes, MK	Long-term Statistics of Response in Non-linear Marine Structures. (Dr.Ing. Thesis)
MTA-90-75	Torbjørn Sotberg, MK	Application of Reliability Methods for Safety Assessment of Submarine Pipelines. (Dr.Ing. Thesis)
MTA-90-76	Zeuthen, Steffen, MP	SEAMAID. A computational model of the design process in a constraint-based logic programming environment. An example from the offshore domain. (Dr.Ing. Thesis)
MTA-91-77	Haagensen, Sven, MM	Fuel Dependant Cyclic Variability in a Spark Ignition Engine - An Optical Approach. (Dr.Ing. Thesis)
MTA-91-78	Løland, Geir, MH	Current forces on and flow through fish farms. (Dr.Ing. Thesis)
MTA-91-79	Hoen, Christopher, MK	System Identification of Structures Excited by Stochastic Load Processes. (Dr.Ing. Thesis)
MTA-91-80	Haugen, Stein, MK	Probabilistic Evaluation of Frequency of Collision between Ships and Offshore Platforms. (Dr.Ing. Thesis)
MTA-91-81	Sødahl, Nils, MK	Methods for Design and Analysis of Flexible Risers. (Dr.Ing. Thesis)
MTA-91-82	Ormberg, Harald, MK	Non-linear Response Analysis of Floating Fish Farm Systems. (Dr.Ing. Thesis)
MTA-91-83	Marley, Mark J., MK	Time Variant Reliability under Fatigue Degradation. (Dr.Ing. Thesis)
MTA-91-84	Krokstad, Jørgen R., MH	Second-order Loads in Multidirectional Seas. (Dr.Ing. Thesis)
MTA-91-85	Molteberg, Gunnar A., MM	The Application of System Identification Techniques to Performance Monitoring of Four Stroke Turbocharged Diesel Engines. (Dr.Ing. Thesis)
MTA-92-86	Mørch, Hans Jørgen Bjelke, MH	Aspects of Hydrofoil Design: with Emphasis on Hydrofoil Interaction in Calm Water. (Dr.Ing. Thesis)
MTA-92-87	Chan Siu Hung, MM	Nonlinear Analysis of Rotordynamic Instabilities in Highspeed Turbomachinery. (Dr.Ing. Thesis)
MTA-92-88	Bessason, Bjarni, MK	Assessment of Earthquake Loading and Response of Seismically Isolated Bridges. (Dr.Ing. Thesis)
MTA-92-89	Langli, Geir, MP	Improving Operational Safety through exploitation of Design Knowledge - an investigation of offshore platform safety. (Dr.Ing. Thesis)
MTA-92-90	Sævik, Svein, MK	On Stresses and Fatigue in Flexible Pipes. (Dr.Ing. Thesis)

MTA-92-91	Ask, Tor Ø., MM	Ignition and Flame Growth in Lean Gas-Air Mixtures. An Experimental Study with a Schlieren System. (Dr.Ing. Thesis)
MTA-86-92	Hessen, Gunnar, MK	Fracture Mechanics Analysis of Stiffened Tubular Members. (Dr.Ing. Thesis)
MTA-93-93	Steinebach, Christian, MM	Knowledge Based Systems for Diagnosis of Rotating Machinery. (Dr.Ing. Thesis)
MTA-93-94	Dalane, Jan Inge, MK	System Reliability in Design and Maintenance of Fixed Offshore Structures. (Dr.Ing. Thesis)
MTA-93-95	Steen, Sverre, MH	Cobblestone Effect on SES. (Dr.Ing. Thesis)
MTA-93-96	Karunakaran, Daniel, MK	Nonlinear Dynamic Response and Reliability Analysis of Drag-dominated Offshore Platforms. (Dr.Ing. Thesis)
MTA-93-97	Hagen, Arnulf, MP	The Framework of a Design Process Language. (Dr.Ing. Thesis)
MTA-93-98	Nordrik, Rune, MM	Investigation of Spark Ignition and Autoignition in Methane and Air Using Computational Fluid Dynamics and Chemical Reaction Kinetics. A Numerical Study of Ignition Processes in Internal Combustion Engines. (Dr.Ing. Thesis)
MTA-94-99	Passano, Elizabeth, MK	Efficient Analysis of Nonlinear Slender Marine Structures. (Dr.Ing. Thesis)
MTA-94-100	Kvålsvold, Jan, MH	Hydroelastic Modelling of Wetdeck Slamming on Multihull Vessels. (Dr.Ing. Thesis)
MTA-94-102	Bech, Sidsel M., MK	Experimental and Numerical Determination of Stiffness and Strength of GRP/PVC Sandwich Structures. (Dr.Ing. Thesis)
MTA-95-103	Paulsen, Hallvard, MM	A Study of Transient Jet and Spray using a Schlieren Method and Digital Image Processing. (Dr.Ing. Thesis)
MTA-95-104	Hovde, Geir Olav, MK	Fatigue and Overload Reliability of Offshore Structural Systems, Considering the Effect of Inspection and Repair. (Dr.Ing. Thesis)
MTA-95-105	Wang, Xiaozhi, MK	Reliability Analysis of Production Ships with Emphasis on Load Combination and Ultimate Strength. (Dr.Ing. Thesis)
MTA-95-106	Ulstein, Tore, MH	Nonlinear Effects of a Flexible Stern Seal Bag on Cobblestone Oscillations of an SES. (Dr.Ing. Thesis)
MTA-95-107	Solaas, Frøydis, MH	Analytical and Numerical Studies of Sloshing in Tanks. (Dr.Ing. Thesis)
MTA-95-108	Hellan, Øyvind, MK	Nonlinear Pushover and Cyclic Analyses in Ultimate Limit State Design and Reassessment of Tubular Steel Offshore Structures. (Dr.Ing. Thesis)
MTA-95-109	Hermundstad, Ole A., MK	Theoretical and Experimental Hydroelastic Analysis of High Speed Vessels. (Dr.Ing. Thesis)

MTA-96-110	Bratland, Anne K., MH	Wave-Current Interaction Effects on Large-Volume Bodies in Water of Finite Depth. (Dr.Ing. Thesis)
MTA-96-111	Herfjord, Kjell, MH	A Study of Two-dimensional Separated Flow by a Combination of the Finite Element Method and Navier-Stokes Equations. (Dr.Ing. Thesis)
MTA-96-112	Æsøy, Vilmar, MM	Hot Surface Assisted Compression Ignition in a Direct Injection Natural Gas Engine. (Dr.Ing. Thesis)
MTA-96-113	Eknes, Monika L., MK	Escalation Scenarios Initiated by Gas Explosions on Offshore Installations. (Dr.Ing. Thesis)
MTA-96-114	Erikstad, Stein O., MP	A Decision Support Model for Preliminary Ship Design. (Dr.Ing. Thesis)
MTA-96-115	Pedersen, Egil, MH	A Nautical Study of Towed Marine Seismic Streamer Cable Configurations. (Dr.Ing. Thesis)
MTA-97-116	Moksnes, Paul O., MM	Modelling Two-Phase Thermo-Fluid Systems Using Bond Graphs. (Dr.Ing. Thesis)
MTA-97-117	Halse, Karl H., MK	On Vortex Shedding and Prediction of Vortex-Induced Vibrations of Circular Cylinders. (Dr.Ing. Thesis)
MTA-97-118	Igland, Ragnar T., MK	Reliability Analysis of Pipelines during Laying, considering Ultimate Strength under Combined Loads. (Dr.Ing. Thesis)
MTA-97-119	Pedersen, Hans-P., MP	Levendefiskteknologi for fiskefartøy. (Dr.Ing. Thesis)
MTA-98-120	Vikestad, Kyrre, MK	Multi-Frequency Response of a Cylinder Subjected to Vortex Shedding and Support Motions. (Dr.Ing. Thesis)
MTA-98-121	Azadi, Mohammad R. E., MK	Analysis of Static and Dynamic Pile-Soil-Jacket Behaviour. (Dr.Ing. Thesis)
MTA-98-122	Ulltang, Terje, MP	A Communication Model for Product Information. (Dr.Ing. Thesis)
MTA-98-123	Torbergsen, Erik, MM	Impeller/Diffuser Interaction Forces in Centrifugal Pumps. (Dr.Ing. Thesis)
MTA-98-124	Hansen, Edmond, MH	A Discrete Element Model to Study Marginal Ice Zone Dynamics and the Behaviour of Vessels Moored in Broken Ice. (Dr.Ing. Thesis)
MTA-98-125	Videiro, Paulo M., MK	Reliability Based Design of Marine Structures. (Dr.Ing. Thesis)
MTA-99-126	Mainçon, Philippe, MK	Fatigue Reliability of Long Welds Application to Titanium Risers. (Dr.Ing. Thesis)
MTA-99-127	Haugen, Elin M., MH	Hydroelastic Analysis of Slamming on Stiffened Plates with Application to Catamaran Wetdecks. (Dr.Ing. Thesis)
MTA-99-128	Langhelle, Nina K., MK	Experimental Validation and Calibration of Nonlinear Finite Element Models for Use in Design

		of Aluminium Structures Exposed to Fire. (Dr.Ing. Thesis)
MTA-99-129	Berstad, Are J., MK	Calculation of Fatigue Damage in Ship Structures. (Dr.Ing. Thesis)
MTA-99-130	Andersen, Trond M., MM	Short Term Maintenance Planning. (Dr.Ing. Thesis)
MTA-99-131	Tveiten, Bård Wathne, MK	Fatigue Assessment of Welded Aluminium Ship Details. (Dr.Ing. Thesis)
MTA-99-132	Søreide, Fredrik, MP	Applications of underwater technology in deep water archaeology. Principles and practice. (Dr.Ing. Thesis)
MTA-99-133	Tønnessen, Rune, MH	A Finite Element Method Applied to Unsteady Viscous Flow Around 2D Blunt Bodies With Sharp Corners. (Dr.Ing. Thesis)
MTA-99-134	Elvekrok, Dag R., MP	Engineering Integration in Field Development Projects in the Norwegian Oil and Gas Industry. The Supplier Management of Norne. (Dr.Ing. Thesis)
MTA-99-135	Fagerholt, Kjetil, MP	Optimeringsbaserte Metoder for Ruteplanlegging innen skipsfart. (Dr.Ing. Thesis)
MTA-99-136	Bysveen, Marie, MM	Visualization in Two Directions on a Dynamic Combustion Rig for Studies of Fuel Quality. (Dr.Ing. Thesis)
MTA-2000-137	Storteig, Eskild, MM	Dynamic characteristics and leakage performance of liquid annular seals in centrifugal pumps. (Dr.Ing. Thesis)
MTA-2000-138	Sagli, Gro, MK	Model uncertainty and simplified estimates of long term extremes of hull girder loads in ships. (Dr.Ing. Thesis)
MTA-2000-139	Tronstad, Harald, MK	Nonlinear analysis and design of cable net structures like fishing gear based on the finite element method. (Dr.Ing. Thesis)
MTA-2000-140	Kroneberg, André, MP	Innovation in shipping by using scenarios. (Dr.Ing. Thesis)
MTA-2000-141	Haslum, Herbjørn Alf, MH	Simplified methods applied to nonlinear motion of spar platforms. (Dr.Ing. Thesis)
MTA-2001-142	Samdal, Ole Johan, MM	Modelling of Degradation Mechanisms and Stressor Interaction on Static Mechanical Equipment Residual Lifetime. (Dr.Ing. Thesis)
MTA-2001-143	Baarholm, Rolf Jarle, MH	Theoretical and experimental studies of wave impact underneath decks of offshore platforms. (Dr.Ing. Thesis)
MTA-2001-144	Wang, Lihua, MK	Probabilistic Analysis of Nonlinear Wave-induced Loads on Ships. (Dr.Ing. Thesis)
MTA-2001-145	Kristensen, Odd H. Holt, MK	Ultimate Capacity of Aluminium Plates under Multiple Loads, Considering HAZ Properties. (Dr.Ing. Thesis)

MTA-2001-146	Greco, Marilena, MH	A Two-Dimensional Study of Green-Water Loading. (Dr.Ing. Thesis)
MTA-2001-147	Heggelund, Svein E., MK	Calculation of Global Design Loads and Load Effects in Large High Speed Catamarans. (Dr.Ing. Thesis)
MTA-2001-148	Babalola, Olusegun T., MK	Fatigue Strength of Titanium Risers – Defect Sensitivity. (Dr.Ing. Thesis)
MTA-2001-149	Mohammed, Abuu K., MK	Nonlinear Shell Finite Elements for Ultimate Strength and Collapse Analysis of Ship Structures. (Dr.Ing. Thesis)
MTA-2002-150	Holmedal, Lars E., MH	Wave-current interactions in the vicinity of the sea bed. (Dr.Ing. Thesis)
MTA-2002-151	Rognebakke, Olav F., MH	Sloshing in rectangular tanks and interaction with ship motions. (Dr.Ing. Thesis)
MTA-2002-152	Lader, Pål Furset, MH	Geometry and Kinematics of Breaking Waves. (Dr.Ing. Thesis)
MTA-2002-153	Yang, Qinzheng, MH	Wash and wave resistance of ships in finite water depth. (Dr.Ing. Thesis)
MTA-2002-154	Melhus, Øyvin, MM	Utilization of VOC in Diesel Engines. Ignition and combustion of VOC released by crude oil tankers. (Dr.Ing. Thesis)
MTA-2002-155	Ronæss, Marit, MH	Wave Induced Motions of Two Ships Advancing on Parallel Course. (Dr.Ing. Thesis)
MTA-2002-156	Økland, Ole D., MK	Numerical and experimental investigation of whipping in twin hull vessels exposed to severe wet deck slamming. (Dr.Ing. Thesis)
MTA-2002-157	Ge, Chunhua, MK	Global Hydroelastic Response of Catamarans due to Wet Deck Slamming. (Dr.Ing. Thesis)
MTA-2002-158	Byklum, Eirik, MK	Nonlinear Shell Finite Elements for Ultimate Strength and Collapse Analysis of Ship Structures. (Dr.Ing. Thesis)
IMT-2003-1	Chen, Haibo, MK	Probabilistic Evaluation of FPSO-Tanker Collision in Tandem Offloading Operation. (Dr.Ing. Thesis)
IMT-2003-2	Skaugset, Kjetil Bjørn, MK	On the Suppression of Vortex Induced Vibrations of Circular Cylinders by Radial Water Jets. (Dr.Ing. Thesis)
IMT-2003-3	Chezhian, Muthu	Three-Dimensional Analysis of Slamming. (Dr.Ing. Thesis)
IMT-2003-4	Buhaug, Øyvind	Deposit Formation on Cylinder Liner Surfaces in Medium Speed Engines. (Dr.Ing. Thesis)
IMT-2003-5	Tregde, Vidar	Aspects of Ship Design: Optimization of Aft Hull with Inverse Geometry Design. (Dr.Ing. Thesis)

IMT-2003-6	Wist, Hanne Therese	Statistical Properties of Successive Ocean Wave Parameters. (Dr.Ing. Thesis)
IMT-2004-7	Ransau, Samuel	Numerical Methods for Flows with Evolving Interfaces. (Dr.Ing. Thesis)
IMT-2004-8	Soma, Torkel	Blue-Chip or Sub-Standard. A data interrogation approach of identity safety characteristics of shipping organization. (Dr.Ing. Thesis)
IMT-2004-9	Ersdal, Svein	An experimental study of hydrodynamic forces on cylinders and cables in near axial flow. (Dr.Ing. Thesis)
IMT-2005-10	Brodtkorb, Per Andreas	The Probability of Occurrence of Dangerous Wave Situations at Sea. (Dr.Ing. Thesis)
IMT-2005-11	Yttervik, Rune	Ocean current variability in relation to offshore engineering. (Dr.Ing. Thesis)
IMT-2005-12	Fredheim, Arne	Current Forces on Net-Structures. (Dr.Ing. Thesis)
IMT-2005-13	Heggernes, Kjetil	Flow around marine structures. (Dr.Ing. Thesis
IMT-2005-14	Fouques, Sebastien	Lagrangian Modelling of Ocean Surface Waves and Synthetic Aperture Radar Wave Measurements. (Dr.Ing. Thesis)
IMT-2006-15	Holm, Håvard	Numerical calculation of viscous free surface flow around marine structures. (Dr.Ing. Thesis)
IMT-2006-16	Bjørheim, Lars G.	Failure Assessment of Long Through Thickness Fatigue Cracks in Ship Hulls. (Dr.Ing. Thesis)
IMT-2006-17	Hansson, Lisbeth	Safety Management for Prevention of Occupational Accidents. (Dr.Ing. Thesis)
IMT-2006-18	Zhu, Xinying	Application of the CIP Method to Strongly Nonlinear Wave-Body Interaction Problems. (Dr.Ing. Thesis)
IMT-2006-19	Reite, Karl Johan	Modelling and Control of Trawl Systems. (Dr.Ing. Thesis)
IMT-2006-20	Smogeli, Øyvind Notland	Control of Marine Propellers. From Normal to Extreme Conditions. (Dr.Ing. Thesis)
IMT-2007-21	Storhaug, Gaute	Experimental Investigation of Wave Induced Vibrations and Their Effect on the Fatigue Loading of Ships. (Dr.Ing. Thesis)
IMT-2007-22	Sun, Hui	A Boundary Element Method Applied to Strongly Nonlinear Wave-Body Interaction Problems. (PhD Thesis, CeSOS)
IMT-2007-23	Rustad, Anne Marthine	Modelling and Control of Top Tensioned Risers. (PhD Thesis, CeSOS)
IMT-2007-24	Johansen, Vegar	Modelling flexible slender system for real-time simulations and control applications

IMT-2007-25	Wroldsen, Anders Sunde	Modelling and control of tensegrity structures. (PhD Thesis, CeSOS)
IMT-2007-26	Aronsen, Kristoffer Høye	An experimental investigation of in-line and combined inline and cross flow vortex induced vibrations. (Dr. avhandling, IMT)
IMT-2007-27	Gao, Zhen	Stochastic Response Analysis of Mooring Systems with Emphasis on Frequency-domain Analysis of Fatigue due to Wide-band Response Processes (PhD Thesis, CeSOS)
IMT-2007-28	Thorstensen, Tom Anders	Lifetime Profit Modelling of Ageing Systems Utilizing Information about Technical Condition. (Dr.ing. thesis, IMT)
IMT-2008-29	Refsnes, Jon Erling Gorset	Nonlinear Model-Based Control of Slender Body AUVs (PhD Thesis, IMT)
IMT-2008-30	Berntsen, Per Ivar B.	Structural Reliability Based Position Mooring. (PhD-Thesis, IMT)
IMT-2008-31	Ye, Naiquan	Fatigue Assessment of Aluminium Welded Box-stiffener Joints in Ships (Dr.ing. thesis, IMT)
IMT-2008-32	Radan, Damir	Integrated Control of Marine Electrical Power Systems. (PhD-Thesis, IMT)
IMT-2008-33	Thomassen, Paul	Methods for Dynamic Response Analysis and Fatigue Life Estimation of Floating Fish Cages. (Dr.ing. thesis, IMT)
IMT-2008-34	Pákozdi, Csaba	A Smoothed Particle Hydrodynamics Study of Two-dimensional Nonlinear Sloshing in Rectangular Tanks. (Dr.ing.thesis, IMT/ CeSOS)
IMT-2007-35	Grytøyr, Guttorm	A Higher-Order Boundary Element Method and Applications to Marine Hydrodynamics. (Dr.ing.thesis, IMT)
IMT-2008-36	Drummen, Ingo	Experimental and Numerical Investigation of Nonlinear Wave-Induced Load Effects in Containerships considering Hydroelasticity. (PhD thesis, CeSOS)
IMT-2008-37	Skejic, Renato	Maneuvering and Seakeeping of a Singel Ship and of Two Ships in Interaction. (PhD-Thesis, CeSOS)
IMT-2008-38	Harlem, Alf	An Age-Based Replacement Model for Repairable Systems with Attention to High-Speed Marine Diesel Engines. (PhD-Thesis, IMT)
IMT-2008-39	Alsos, Hagbart S.	Ship Grounding. Analysis of Ductile Fracture, Bottom Damage and Hull Girder Response. (PhD-thesis, IMT)
IMT-2008-40	Graczyk, Mateusz	Experimental Investigation of Sloshing Loading and Load Effects in Membrane LNG Tanks Subjected to Random Excitation. (PhD-thesis, CeSOS)
IMT-2008-41	Taghipour, Reza	Efficient Prediction of Dynamic Response for Flexible amd Multi-body Marine Structures. (PhD-thesis, CeSOS)

IMT-2008-42	Ruth, Eivind	Propulsion control and thrust allocation on marine vessels. (PhD thesis, CeSOS)
IMT-2008-43	Nystad, Bent Helge	Technical Condition Indexes and Remaining Useful Life of Aggregated Systems. PhD thesis, IMT
IMT-2008-44	Soni, Prashant Kumar	Hydrodynamic Coefficients for Vortex Induced Vibrations of Flexible Beams, PhD thesis, CeSOS
IMT-2009-45	Amlashi, Hadi K.K.	Ultimate Strength and Reliability-based Design of Ship Hulls with Emphasis on Combined Global and Local Loads. PhD Thesis, IMT
IMT-2009-46	Pedersen, Tom Arne	Bond Graph Modelling of Marine Power Systems. PhD Thesis, IMT
IMT-2009-47	Kristiansen, Trygve	Two-Dimensional Numerical and Experimental Studies of Piston-Mode Resonance. PhD-Thesis, CeSOS
IMT-2009-48	Ong, Muk Chen	Applications of a Standard High Reynolds Number Model and a Stochastic Scour Prediction Model for Marine Structures. PhD-thesis, IMT
IMT-2009-49	Hong, Lin	Simplified Analysis and Design of Ships subjected to Collision and Grounding. PhD-thesis, IMT
IMT-2009-50	Koushan, Kamran	Vortex Induced Vibrations of Free Span Pipelines, PhD thesis, IMT
IMT-2009-51	Korsvik, Jarl Eirik	Heuristic Methods for Ship Routing and Scheduling. PhD-thesis, IMT
IMT-2009-52	Lee, Jihoon	Experimental Investigation and Numerical in Analyzing the Ocean Current Displacement of Longlines. Ph.d.-Thesis, IMT.
IMT-2009-53	Vestbøstad, Tone Gran	A Numerical Study of Wave-in-Deck Impact usin a Two-Dimensional Constrained Interpolation Profile Method, Ph.d.thesis, CeSOS.
IMT-2009-54	Bruun, Kristine	Bond Graph Modelling of Fuel Cells for Marine Power Plants. Ph.d.-thesis, IMT
IMT 2009-55	Holstad, Anders	Numerical Investigation of Turbulence in a Sekwed Three-Dimensional Channel Flow, Ph.d.-thesis, IMT.
IMT 2009-56	Ayala-Uraga, Efren	Reliability-Based Assessment of Deteriorating Ship-shaped Offshore Structures, Ph.d.-thesis, IMT
IMT 2009-57	Kong, Xiangjun	A Numerical Study of a Damaged Ship in Beam Sea Waves. Ph.d.-thesis, IMT/CeSOS.
IMT 2010-58	Kristiansen, David	Wave Induced Effects on Floaters of Aquaculture Plants, Ph.d.-thesis, CeSOS.
IMT 2010-59	Ludvigsen, Martin	An ROV-Toolbox for Optical and Acoustic Scientific Seabed Investigation. Ph.d.-thesis IMT.

IMT 2010-60	Hals, Jørgen	Modelling and Phase Control of Wave-Energy Converters. Ph.d.thesis, CeSOS.
IMT 2010- 61	Shu, Zhi	Uncertainty Assessment of Wave Loads and Ultimate Strength of Tankers and Bulk Carriers in a Reliability Framework. Ph.d. Thesis, IMT/ CeSOS
IMT 2010-62	Shao, Yanlin	Numerical Potential-Flow Studies on Weakly-Nonlinear Wave-Body Interactions with/without Small Forward Speed, Ph.d.thesis,CeSOS.
IMT 2010-63	Califano, Andrea	Dynamic Loads on Marine Propellers due to Intermittent Ventilation. Ph.d.thesis, IMT.
IMT 2010-64	El Khoury, George	Numerical Simulations of Massively Separated Turbulent Flows, Ph.d.-thesis, IMT
IMT 2010-65	Seim, Knut Sponheim	Mixing Process in Dense Overflows with Emphasis on the Faroe Bank Channel Overflow. Ph.d.thesis, IMT
IMT 2010-66	Jia, Huirong	Structural Analysis of Intect and Damaged Ships in a Collission Risk Analysis Perspective. Ph.d.thesis CeSoS.
IMT 2010-67	Jiao, Linlin	Wave-Induced Effects on a Pontoon-type Very Large Floating Structures (VLFS). Ph.D.-thesis, CeSOS.
IMT 2010-68	Abrahamsen, Bjørn Christian	Sloshing Induced Tank Roof with Entrapped Air Pocket. Ph.d.thesis, CeSOS.
IMT 2011-69	Karimirad, Madjid	Stochastic Dynamic Response Analysis of Spar-Type Wind Turbines with Catenary or Taut Mooring Systems. Ph.d.-thesis, CeSOS.
IMT - 2011-70	Erlend Meland	Condition Monitoring of Safety Critical Valves. Ph.d.-thesis, IMT.
IMT – 2011-71	Yang, Limin	Stochastic Dynamic System Analysis of Wave Energy Converter with Hydraulic Power Take-Off, with Particular Reference to Wear Damage Analysis, Ph.d. Thesis, CeSOS.
IMT – 2011-72	Visscher, Jan	Application of Particla Image Velocimetry on Turbulent Marine Flows, Ph.d.Thesis, IMT.
IMT – 2011-73	Su, Biao	Numerical Predictions of Global and Local Ice Loads on Ships. Ph.d.Thesis, CeSOS.
IMT – 2011-74	Liu, Zhenhui	Analytical and Numerical Analysis of Iceberg Collision with Ship Structures. Ph.d.Thesis, IMT.
IMT – 2011-75	Aarsæther, Karl Gunnar	Modeling and Analysis of Ship Traffic by Observation and Numerical Simulation. Ph.d.Thesis, IMT.

Imt – 2011-76	Wu, Jie	Hydrodynamic Force Identification from Stochastic Vortex Induced Vibration Experiments with Slender Beams. Ph.d.Thesis, IMT.
Imt – 2011-77	Amini, Hamid	Azimuth Propulsors in Off-design Conditions. Ph.d.Thesis, IMT.
IMT – 2011-78	Nguyen, Tan-Hoi	Toward a System of Real-Time Prediction and Monitoring of Bottom Damage Conditions During Ship Grounding. Ph.d.thesis, IMT.
IMT-2011-79	Tavakoli, Mohammad T.	Assessment of Oil Spill in Ship Collision and Grounding, Ph.d.thesis, IMT.
IMT-2011-80	Guo, Bingjie	Numerical and Experimental Investigation of Added Resistance in Waves. Ph.d.Thesis, IMT.
IMT-2011-81	Chen, Qiaofeng	Ultimate Strength of Aluminium Panels, considering HAZ Effects, IMT
IMT-2012-82	Kota, Ravikiran S.	Wave Loads on Decks of Offshore Structures in Random Seas, CeSOS.
IMT-2012-83	Sten, Ronny	Dynamic Simulation of Deep Water Drilling Risers with Heave Compensating System, IMT.
IMT-2012-84	Berle, Øyvind	Risk and resilience in global maritime supply chains, IMT.
IMT-2012-85	Fang, Shaoji	Fault Tolerant Position Mooring Control Based on Structural Reliability, CeSOS.
IMT-2012-86	You, Jikun	Numerical studies on wave forces and moored ship motions in intermediate and shallow water, CeSOS.
IMT-2012-87	Xiang ,Xu	Maneuvering of two interacting ships in waves, CeSOS
IMT-2012-88	Dong, Wenbin	Time-domain fatigue response and reliability analysis of offshore wind turbines with emphasis on welded tubular joints and gear components, CeSOS
IMT-2012-89	Zhu, Suji	Investigation of Wave-Induced Nonlinear Load Effects in Open Ships considering Hull Girder Vibrations in Bending and Torsion, CeSOS
IMT-2012-90	Zhou, Li	Numerical and Experimental Investigation of Station-keeping in Level Ice, CeSOS
IMT-2012-91	Ushakov, Sergey	Particulate matter emission characteristics from diesel enignes operating on conventional and alternative marine fuels, IMT

IMT-2013-1	Yin, Decao	Experimental and Numerical Analysis of Combined In-line and Cross-flow Vortex Induced Vibrations, CeSOS
IMT-2013-2	Kurniawan, Adi	Modelling and geometry optimisation of wave energy converters, CeSOS
IMT-2013-3	Al Ryati, Nabil	Technical condition indexes doe auxiliary marine diesel engines, IMT
IMT-2013-4	Firoozkoohi, Reza	Experimental, numerical and analytical investigation of the effect of screens on sloshing, CeSOS
IMT-2013-5	Ommani, Babak	Potential-Flow Predictions of a Semi-Displacement Vessel Including Applications to Calm Water Broaching, CeSOS
IMT-2013-6	Xing, Yihan	Modelling and analysis of the gearbox in a floating spar-type wind turbine, CeSOS
IMT-7-2013	Balland, Océane	Optimization models for reducing air emissions from ships, IMT
IMT-8-2013	Yang, Dan	Transitional wake flow behind an inclined flat plate-----Computation and analysis, IMT
IMT-9-2013	Abdillah, Suyuthi	Prediction of Extreme Loads and Fatigue Damage for a Ship Hull due to Ice Action, IMT
IMT-10-2013	Ramìrez, Pedro Agustìn Pèrez	Ageing management and life extension of technical systems- Concepts and methods applied to oil and gas facilities, IMT
IMT-11-2013	Chuang, Zhenju	Experimental and Numerical Investigation of Speed Loss due to Seakeeping and Maneuvering. IMT
IMT-12-2013	Etemaddar, Mahmoud	Load and Response Analysis of Wind Turbines under Atmospheric Icing and Controller System Faults with Emphasis on Spar Type Floating Wind Turbines, IMT
IMT-13-2013	Lindstad, Haakon	Strategies and measures for reducing maritime CO2 emissons, IMT
IMT-14-2013	Haris, Sabril	Damage interaction analysis of ship collisions, IMT
IMT-15-2013	Shainee, Mohamed	Conceptual Design, Numerical and Experimental Investigation of a SPM Cage Concept for Offshore Mariculture, IMT
IMT-16-2013	Gansel, Lars	Flow past porous cylinders and effects of biofouling and fish behavior on the flow in and around Atlantic salmon net cages, IMT
IMT-17-2013	Gaspar, Henrique	Handling Aspects of Complexity in Conceptual Ship Design, IMT
IMT-18-2013	Thys, Maxime	Theoretical and Experimental Investigation of a Free Running Fishing Vessel at Small Frequency of Encounter, CeSOS

IMT-19-2013	Aglen, Ida	VIV in Free Spanning Pipelines, CeSOS
IMT-1-2014	Song, An	Theoretical and experimental studies of wave diffraction and radiation loads on a horizontally submerged perforated plate, CeSOS
IMT-2-2014	Rogne, Øyvind Ygre	Numerical and Experimental Investigation of a Hinged 5-body Wave Energy Converter, CeSOS
IMT-3-2014	Dai, Lijuan	Safe and efficient operation and maintenance of offshore wind farms ,IMT
IMT-4-2014	Bachynski, Erin Elizabeth	Design and Dynamic Analysis of Tension Leg Platform Wind Turbines, CeSOS
IMT-5-2014	Wang, Jingbo	Water Entry of Freefall Wedged – Wedge motions and Cavity Dynamics, CeSOS
IMT-6-2014	Kim, Ekaterina	Experimental and numerical studies related to the coupled behavior of ice mass and steel structures during accidental collisions, IMT
IMT-7-2014	Tan, Xiang	Numerical investigation of ship's continuous- mode icebreaking in leverl ice, CeSOS
IMT-8-2014	Muliawan, Made Jaya	Design and Analysis of Combined Floating Wave and Wind Power Facilities, with Emphasis on Extreme Load Effects of the Mooring System, CeSOS
IMT-9-2014	Jiang, Zhiyu	Long-term response analysis of wind turbines with an emphasis on fault and shutdown conditions, IMT
IMT-10-2014	Dukan, Fredrik	ROV Motion Control Systems, IMT
IMT-11-2014	Grimsmo, Nils I.	Dynamic simulations of hydraulic cylinder for heave compensation of deep water drilling risers, IMT
IMT-12-2014	Kvittem, Marit I.	Modelling and response analysis for fatigue design of a semisubmersible wind turbine, CeSOS
IMT-13-2014	Akhtar, Juned	The Effects of Human Fatigue on Risk at Sea, IMT
IMT-14-2014	Syahroni, Nur	Fatigue Assessment of Welded Joints Taking into Account Effects of Residual Stress, IMT
IMT-1-2015	Bøckmann, Eirik	Wave Propulsion of ships, IMT
IMT-2-2015	Wang, Kai	Modelling and dynamic analysis of a semi-submersible floating vertical axis wind turbine, CeSOS
IMT-3-2015	Fredriksen, Arnt Gunvald	A numerical and experimental study of a two-dimensional body with moonpool in waves and current, CeSOS
IMT-4-2015	Jose Patricio Gallardo Canabes	Numerical studies of viscous flow around bluff bodies, IMT

IMT-5-2015	Vegard Longva	Formulation and application of finite element techniques for slender marine structures subjected to contact interactions, IMT
IMT-6-2015	Jacobus De Vaal	Aerodynamic modelling of floating wind turbines, CeSOS
IMT-7-2015	Fachri Nasution	Fatigue Performance of Copper Power Conductors, IMT
IMT-8-2015	Oleh I Karpa	Development of bivariate extreme value distributions for applications in marine technology,CeSOS
IMT-9-2015	Daniel de Almeida Fernandes	An output feedback motion control system for ROVs, AMOS
IMT-10-2015	Bo Zhao	Particle Filter for Fault Diagnosis: Application to Dynamic Positioning Vessel and Underwater Robotics, CeSOS
IMT-11-2015	Wenting Zhu	Impact of emission allocation in maritime transportation, IMT
IMT-12-2015	Amir Rasekhi Nejad	Dynamic Analysis and Design of Gearboxes in Offshore Wind Turbines in a Structural Reliability Perspective, CeSOS
IMT-13-2015	Arturo Jesùs Ortega Malca	Dynamic Response of Flexibles Risers due to Unsteady Slug Flow, CeSOS
IMT-14-2015	Dagfinn Husjord	Guidance and decision-support system for safe navigation of ships operating in close proximity, IMT
IMT-15-2015	Anirban Bhattacharyya	Ducted Propellers: Behaviour in Waves and Scale Effects, IMT
IMT-16-2015	Qin Zhang	Image Processing for Ice Parameter Identification in Ice Management, IMT
IMT-1-2016	Vincentius Rumawas	Human Factors in Ship Design and Operation: An Experiential Learning, IMT
IMT-2-2016	Martin Storheim	Structural response in ship-platform and ship-ice collisions, IMT
IMT-3-2016	Mia Abrahamsen Prsic	Numerical Simulations of the Flow around single and Tandem Circular Cylinders Close to a Plane Wall, IMT
IMT-4-2016	Tufan Arslan	Large-eddy simulations of cross-flow around ship sections, IMT
IMT-5-2016	Pierre Yves-Henry	Parametrisation of aquatic vegetation in hydraulic and coastal research,IMT

IMT-6-2016	Lin Li	Dynamic Analysis of the Instalation of Monopiles for Offshore Wind Turbines, CeSOS
IMT-7-2016	Øivind Kåre Kjerstad	Dynamic Positioning of Marine Vessels in Ice, IMT
IMT-8-2016	Xiaopeng Wu	Numerical Analysis of Anchor Handling and Fish Trawling Operations in a Safety Perspective, CeSOS
IMT-9-2016	Zhengshun Cheng	Integrated Dynamic Analysis of Floating Vertical Axis Wind Turbines, CeSOS
IMT-10-2016	Ling Wan	Experimental and Numerical Study of a Combined Offshore Wind and Wave Energy Converter Concept
IMT-11-2016	Wei Chai	Stochastic dynamic analysis and reliability evaluation of the roll motion for ships in random seas, CeSOS
IMT-12-2016	Øyvind Selnes Patricksson	Decision support for conceptual ship design with focus on a changing life cycle and future uncertainty, IMT
IMT-13-2016	Mats Jørgen Thorsen	Time domain analysis of vortex-induced vibrations, IMT
IMT-14-2016	Edgar McGuinness	Safety in the Norwegian Fishing Fleet – Analysis and measures for improvement, IMT
IMT-15-2016	Sepideh Jafarzadeh	Energy effiency and emission abatement in the fishing fleet, IMT
IMT-16-2016	Wilson Ivan Guachamin Acero	Assessment of marine operations for offshore wind turbine installation with emphasis on response-based operational limits, IMT
IMT-17-2016	Mauro Candeloro	Tools and Methods for Autonomous Operations on Seabed and Water Coumn using Underwater Vehicles, IMT
IMT-18-2016	Valentin Chabaud	Real-Time Hybrid Model Testing of Floating Wind Tubines, IMT
IMT-1-2017	Mohammad Saud Afzal	Three-dimensional streaming in a sea bed boundary layer
IMT-2-2017	Peng Li	A Theoretical and Experimental Study of Wave-induced Hydroelastic Response of a Circular Floating Collar
IMT-3-2017	Martin Bergström	A simulation-based design method for arctic maritime transport systems
IMT-4-2017	Bhushan Taskar	The effect of waves on marine propellers and propulsion
IMT-5-2017	Mohsen Bardestani	A two-dimensional numerical and experimental study of a floater with net and sinker tube in waves and current

IMT-6-2017	Fatemeh Hoseini Dadmarzi	Direct Numerical Simualtion of turbulent wakes behind different plate configurations
IMT-7-2017	Michel R. Miyazaki	Modeling and control of hybrid marine power plants
IMT-8-2017	Giri Rajasekhar Gunnu	Safety and effiency enhancement of anchor handling operations with particular emphasis on the stability of anchor handling vessels
IMT-9-2017	Kevin Koosup Yum	Transient Performance and Emissions of a Turbocharged Diesel Engine for Marine Power Plants
IMT-10-2017	Zhaolong Yu	Hydrodynamic and structural aspects of ship collisions
IMT-11-2017	Martin Hassel	Risk Analysis and Modelling of Allisions between Passing Vessels and Offshore Installations
IMT-12-2017	Astrid H. Brodtkorb	Hybrid Control of Marine Vessels – Dynamic Positioning in Varying Conditions
IMT-13-2017	Kjersti Bruserud	Simultaneous stochastic model of waves and current for prediction of structural design loads
IMT-14-2017	Finn-Idar Grøtta Giske	Long-Term Extreme Response Analysis of Marine Structures Using Inverse Reliability Methods
IMT-15-2017	Stian Skjong	Modeling and Simulation of Maritime Systems and Operations for Virtual Prototyping using co-Simulations
IMT-1-2018	Yingguang Chu	Virtual Prototyping for Marine Crane Design and Operations
IMT-2-2018	Sergey Gavrilin	Validation of ship manoeuvring simulation models
IMT-3-2018	Jeevith Hegde	Tools and methods to manage risk in autonomous subsea inspection,maintenance and repair operations
IMT-4-2018	Ida M. Strand	Sea Loads on Closed Flexible Fish Cages
IMT-5-2018	Erlend Kvinge Jørgensen	Navigation and Control of Underwater Robotic Vehicles
IMT-6-2018	Bård Stovner	Aided Intertial Navigation of Underwater Vehicles
IMT-7-2018	Erlend Liavåg Grotle	Thermodynamic Response Enhanced by Sloshing in Marine LNG Fuel Tanks
IMT-8-2018	Børge Rokseth	Safety and Verification of Advanced Maritime Vessels

IMT-9-2018	Jan Vidar Ulveseter	Advances in Semi-Empirical Time Domain Modelling of Vortex-Induced Vibrations
IMT-10-2018	Chenyu Luan	Design and analysis for a steel braceless semi-submersible hull for supporting a 5-MW horizontal axis wind turbine
IMT-11-2018	Carl Fredrik Rehn	Ship Design under Uncertainty
IMT-12-2018	Øyvind Ødegård	Towards Autonomous Operations and Systems in Marine Archaeology
IMT-13-2018	Stein Melvær Nornes	Guidance and Control of Marine Robotics for Ocean Mapping and Monitoring
IMT-14-2018	Petter Norgren	Autonomous Underwater Vehicles in Arctic Marine Operations: Arctic marine research and ice monitoring
IMT-15-2018	Minjoo Choi	Modular Adaptable Ship Design for Handling Uncertainty in the Future Operating Context
MT-16-2018	Ole Alexander Eidsvik	Dynamics of Remotely Operated Underwater Vehicle Systems
IMT-17-2018	Mahdi Ghane	Fault Diagnosis of Floating Wind Turbine Drivetrain- Methodologies and Applications
IMT-18-2018	Christoph Alexander Thieme	Risk Analysis and Modelling of Autonomous Marine Systems