Journal of Marine Science and Technology (2023) 28:624–648
https://doi.org/10.1007/s00773-023-00945-6

ORIGINAL ARTICLE

Risk-based supervisory control for autonomous ship navigation

Simon Blindheim[1] · **Tor Arne Johansen**[1] · **Ingrid Bouwer Utne**[2]

Received: 30 June 2022 / Accepted: 25 May 2023 / Published online: 9 June 2023

Abstract
This paper proposes a novel method to transform the results of qualitative risk analysis into a numeric optimal control problem for autonomous ship navigation. Today, making autonomous high-level decisions replacing a crew onboard is considered difficult, in some part due to the complexity of managing the operational risks involved. Although human supervisors, e.g., located in remote operating control centers are still needed for safety and liability reasons, there is a growing demand for complex decisions to be made by the onboard control system itself, both during normal operations and in emergencies. This paper suggests general principles for how the results from systems-theoretic process analysis (STPA) can be transformed into a quantitative and computationally tractable optimization problem, solved by a MPC-based decision-making algorithm for autonomous navigation. The proposed method is demonstrated and evaluated by simulating an autonomous ship navigating in a coastal environment. It is concluded that the proposed method may serve as a reasonable and valuable bridge between the realms of qualitative risk analysis and numerical optimal control for risk-aware autonomous control and decision-making.

Keywords Accident prevention · Autonomous control · Navigation · Model predictive control (MPC) · Online consequence analysis · Optimal control · Risk assessment · Risk control · Real-time decision-making · Systems-theoretic process analysis (STPA) · Trajectory planning

Abbreviations

AMM	Autonomous machinery management ANS autonomous navigation system
BBN	Bayesian belief network
CC	Controller constraints
COLREGs	The International Regulations for Preventing Collisions at Sea
DP	Dynamic positioning
ENC	Electronic navigational charts
LQR	Linear-quadratic regulator
MPC	Model predictive control
NE	North-East
NLP	Nonlinear programming
OCP	Optimal control problem
ORE	Online risk estimation
PSO	Particle swarm optimization
ROC	Remote operations center
RPN	Risk priority numbers
SAS	Situational awareness system
SC	Safety constraints
SCA	Supervisory control actions
SI	Safety inequalities
STAMP	Systems-Theoretic Accident Model and Processes
STPA	System-Theoretic Process Analysis SV slack variables
UCA	Unsafe supervisory control actions

1 Introduction

There is an increasing desire to reduce operational costs and risks during ship operations by improving the intelligence and autonomous decision-making capabilities of maritime vessels [1]. Bridging the gap, however, between qualitative risk analysis and quantitative supervisory optimal control is a challenging task. The aim of this work is to develop a method for applying results from risk analysis to be utilized by an supervisory optimal control algorithm. Results from

✉ Simon Blindheim
simon.blindheim@ntnu.no

[1] Department of Engineering Cybernetics, Centre for Autonomous Marine Operations and Systems, Norwegian University of Science and Technology, 7491 Trondheim, Norway

[2] Department of Marine Technology, Centre for Autonomous Marine Operations and Systems, Norwegian University of Science and Technology, 7052 Trondheim, Norway

risk analysis may provide useful input to determining safe and efficient sequences of control actions to be taken in a complex maritime environment.

Higher levels of autonomy are not the main objective in itself, but rather to realize safer and more efficient operations involving human personnel. One way of improving *human safety* may be to move operators to an remote operations center (ROC), which may increase the productivity and efficiency of operations by giving, e.g., ship captains the opportunity to focus their abilities on monitoring a larger fleet of vessels simultaneously, supported by increased analysis of human factors or interference related to the altered supervision and (semi-)autonomous control hierarchy [2, 3].

Caution should nonetheless be exercised when making changes to a large operational infrastructure such as cargo or passenger transport. Communication and cooperation with conventional ships and compliance with international regulations, such as the International Regulations for Preventing Collisions at Sea (COLREGs) are decisive. The safety and well-being of smaller vessels also need to be taken into account, e.g., smaller sailboats sailing in the vicinity of the larger (semi-)autonomous vessels.

In general, accidents occur due to unpredictable conditions, erroneous decision-making, or unexpected emergent system failures [4–6]. Risk assessment is therefore required to identify and analyze hazardous events, and determine the need for potential risk mitigation measures. One potential measure for autonomous ships is to implement onboard online consequence analysis-based optimization algorithms with some prediction horizon, weighting different operational objectives in light of the risks associated with each action considered for execution. Model predictive control (MPC) MPC is such a method, and has shown promising results for development of operational constraints [7], dynamic positioning (DP) [8], path following [9] and collision avoidance [10, 11].

These systems, however, usually have strictly defined operational areas or limited available decision spaces in which they are explicitly allowed to make autonomous decisions. The conditions are normally the default operational stages, like the crossing transit phase of an autonomous ferry. Nor do typical applications of MPC include risk models or results from risk analysis as input to the optimization algorithm. In order to reach even higher levels of autonomy, a high-level supervisory control system for risk assessment and safety-aware decision-making is needed [12].

System-Theoretic Process Analysis (STPA) considers safety as a control problem, which makes it feasible for revealing hazards related to autonomous systems. Such hazards should be considered in the design of control algorithms and when optimizing decisions during operations to improve safety. One of the challenges with STPA, however, is that it only brings forward qualitative results, which are impossible to use directly for MPC.

In this paper, a step-by-step approach to design of supervisory risk control and risk-aware MPC is proposed as follows: (i) Risk analysis or hazard identification in terms of a STPA is performed in order to identify how inadequate control of a maritime vessel may occur. Next, (ii) the qualitative results of the STPA are transformed into an optimal control problem (OCP), subsequently (iii) solved numerically by nonlinear programming through an existing MPC-based decision-making algorithm for path planning with anti-grounding [13]. The ship control is performed using a receding horizon approach, based on the chosen dynamic ship model and a combination of cost functions and operational limitations, each targeting different aspects of online path planning and risk management. The performance of the developed risk terms in the MPC cost function is furthermore demonstrated by example simulations.

Ultimately, the novel contribution of this paper is a method for transforming the results of qualitative risk analysis into a tractable optimization problem to be solved by an online decision-making algorithm. It provides a systematic approach to the design of nonlinear MPC cost, constraints, and solution strategy, with systematic considerations for hazards and risks, which is a highly challenging task. The resulting framework may serve as a foundation for future autonomous decision-making or online consequence analysis techniques for both accident prevention and online risk control. To the authors' knowledge, this is the first time qualitative risk analysis and MPC are explicitly coupled.

2 Background

2.1 Supervisory risk control

Risk analysis in general is concerned with identifying what may go wrong, and determining the likelihoods and consequences of those events [14]. Risk modeling represents risk qualitatively or quantitatively, and risk control is the use and integration of such models to support situation awareness and decision-making, e.g., by autonomous systems [12]. Separating the performed risk control into two equally important and dependent "modes", i.e., by human supervision of an automated or autonomous system, and by an autonomous system itself, supervisory risk control focuses on the latter of the two [15].

The control of an autonomous system may be divided into the offline mission planner layer, the online guidance and optimization level, and the control execution level [16]. The work in this paper is mainly focused on control related to guidance and optimization, for which the high-level mission

is given and preplanned, and the lower-level control is taken care of by corresponding subsystems.

Previous work within supervisory risk control proposes to use Bayesian belief network (BBN) for *online risk control* [12], in which the BBN serves as an underlying risk model in order to update and assess the current risk levels during operations. This is in contrast to the approach presented in this paper, wherein the main objective is to use the results of the risk analysis in the design process of the online control algorithm. Though both methods involve updating risk levels during runtime, no underlying risk model is used in this work. Instead, a risk-based cost function is constructed and used to solve the resulting numeric OCP.

2.2 STAMP and STPA

The Systems-Theoretic Accident Model and Processes (STAMP) [17] is an accident causation model for analyzing and explaining how accidents may occur, attempting to handle the ever-increasing complexity of systems. Complex systems may have emergent properties only surfacing through the interconnection of the various parts of the system, which is difficult to predict by component failure analysis alone. STAMP regards safety as a control problem, preventing accidents by controlling the system or process according to appropriate safety constraints (SC). Accidents may thus occur if these constraints are broken, i.e., inadequate control. STPA [17] is a hazard analysis method based on STAMP, which attempts to identify how hazards or inadequate control may take place. This method is feasible for complex and automated maritime control systems [18] and as a basis for safety verification [19], and is generally applied through the following steps:

1. Define losses, system-level hazards and system-level safety constraints.
2. Define a system representation by analyzing and modeling the system as a hierarchical control structure.
3. Identify unsafe control actions.
4. Identify loss scenarios in which the unsafe control actions may occur.

The first step identifies or defines what types of losses one wants to prevent. In the case of autonomous ships, one may specify what the focus of the analysis is aimed at, e.g., fires, grounding, collision, or security threats such as piracy. Thus, the scope and purpose of the analysis is clarified by defining how the system-level hazardous states may lead to such losses. Next, the system is modeled and represented as a hierarchical control structure, i.e., a set of feedback control loops, and all relevant process model variables related to the internal belief of the controllers are identified. The third step is to identify unsafe control actions, i.e., supervisory control actions (SCA) that in any way may lead to one of the system-level hazards.

Note that supervisory control actions in the context of this risk analysis are defined as any type of decision or program flow between system modules which allows for hazardous states or outcomes to develop or occur, in contrast to automatic control actions typically performed by machinery systems (i.e., propulsion or steering control). STPA uses four general categories of unsafe control actions, which are presented in Table 1 [20].

Scenarios in which unsafe control actions may occur and their causes are identified by inspecting relevant parts of the control loops in the control hierarchy in specific contexts, e.g., incorrect feedback, lack of feedback, decision-making flaws, time-delay, (lack of) situation awareness, component failures, process disturbances, communication errors, or other risk influencing factors related to the control loops.

The identified scenarios from the STPA may subsequently be considered for testing and simulation, and the results used to construct a suitable risk-based cost function for the MPC-based decision-making algorithm.

2.3 Scope and simplifications

An important prerequisite for risk analysis is to define its scope. The main objective is to develop an optimization-based algorithm controlling the ship, given a preplanned path in a maritime environment. To ensure safe optimization and that hazards and risks are considered in the optimization, STPA is performed and results are transformed into mathematical constraints, logic and objectives, implemented in MPC. The results from STPA are used as input to construct an OCP, subsequently discretized into a nonlinear program and solved by an MPC algorithm, and assessing the resulting trajectories. The goal is to improve the autonomy of the control system, and enhance safe operation of autonomous ships.

The cost function of the resulting MPC consists of risk-based cost terms. Unlike the established nomenclature of shipping economics concerned with e.g., capital and operational expenditures [21], the concept of cost in this context is specifically applied solely to weigh and balance opposing interests or penalties as part of the standard terminology used within the field of optimization.

Table 1 Categories of unsafe control actions

A	A SCA required for safety is not provided or not followed
B	A SCA that causes a hazard is provided
C	A SCA is provided too early, too late or in the wrong sequence
D	A SCA is applied for a too long or too short period of time

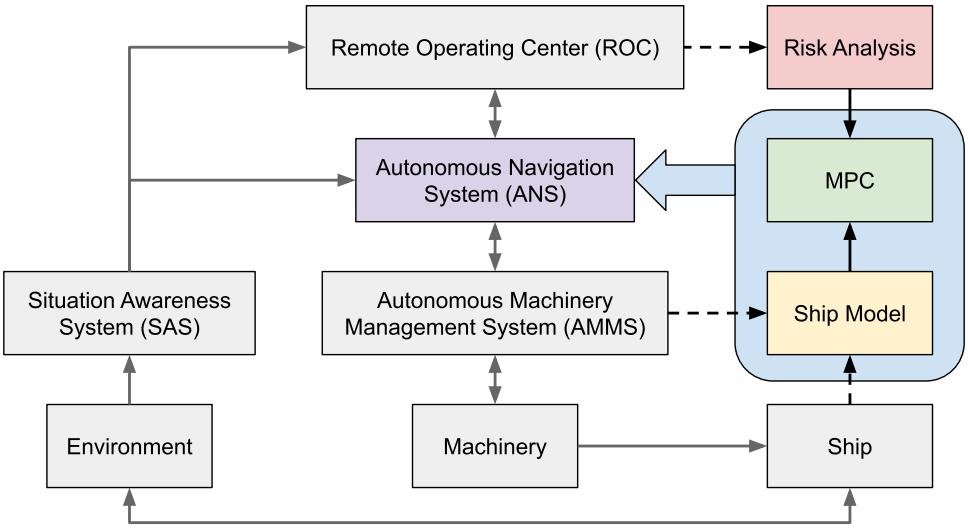

Fig. 1 Scope of the system considered in this work

A simplified overview of the system scope of this work is shown in Fig. 1. The implemented software used for simulations is highlighted in blue, and represents a core element of a simulated ANS. The MPC algorithm utilizes a mathematical *Ship Model* (the dynamical model in the Appendix) to predict future states given the current system state. This structure assumes communications between systems such as a situational awareness system (SAS) and an autonomous machinery management (AMM) system, controlling the ship's machinery. These system modules shown in gray color are, however, simplified in this work to only contain and relay their intended inputs and outputs between appropriate modules of the implemented software.

This paper is mainly concerned with grounding hazards, allision, and anti-grounding functionality, for proof of concept. Thus, collision avoidance with respect to dynamic obstacles, e.g., other maritime vessels, is not considered in this work. Disturbances applied to the system are simplified such that only wind direction and wind velocity is considered, i.e., no current or wave disturbances are included. Furthermore, COLREGs compliance (or violation) is disregarded. As the act of docking (or berthing) the autonomous ship may be viewed as a separate control mode, docking is not considered in this work. The approach in the paper assumes that the autonomous ship is able to execute appropriate emergency measures or otherwise surrender control of the ship to human operators if the supervisory risk control algorithm exceeds a certain risk threshold or enters a specific hazardous situation. Additionally, consequences related to or occurring *as a result* of grounding, such as environmental pollution or loss of human lives, are not included in the risk analysis. This is considered appropriate for a proof-of-concept study.

Even though approximations are used, it is proposed that extensions for more complex analysis and risk modeling, such as collision avoidance and COLREGs compliance, may be equivalently added using the presented method in future works, without loss of generality.

Table 2 Accidents

A1	Allision with a stationary mapped obstacle
A2	The ship grounds or makes contact with the seafloor

Table 3 System-level hazards

H1	The ship violates the minimum separation distance to a stationary obstacle (**A1**)
H2	The ship violates the minimum separation distance to the shore (**A2**)
H3	The ship sails in too shallow water (**A2**)

3 Modeling

The STPA in this paper is based on an analysis performed in workshops together with industry participants during Spring 2019. The main objectives of the control process is to: (a) to avoid grounding and allision with mapped obstacles, and (b) complete the given mission of the ship operations with optimized resource allocation weighted against risk considerations from (a), within defined limits.

3.1 STPA Step 1: Purpose

Step 1 of the STPA is presented in Tables 2, 3, 4 and 5, to be used as a basis for the following steps. The hazards were specified with respect to which motion control objectives

Table 4 System-level safety constraints

SSC1	The minimum separation distance to obstacles must be maintained
SSC2	The minimum separation distance to shore must be maintained
SSC3	The ship must not sail in too shallow water

that may lead to violation of the safety constraints, defined in Table 5.

3.2 STPA Step 2: System representation

The system control flow and feedback hierarchy of this work is shown in Fig. 2, and consists of all implemented software modules and physical entities present in the system, including peripheral interfaces between the software system and the physical environment. Directed connections between the various modules show the program flow.

Ship denotes the physical ship, which also encompasses all software modules communicating with the *ANS*, as well as the presence of a physical ship hull and machinery interacting with the physical environment. Numerical values for physical *Environment* forces and states are provided to the Ship by the *SAS* module through forecasts and sensor measurements, which are in turn interpreted by the *MPC* through its *Scenario* module. Additionally, the *Dynamics* module contains the ship's equations of motion, which are used to generate simulated (expected) movement within the physical environment.

These system states along with hypsometry data from the *ENC* module [22] are utilized by the Supervisory Risk Control module to predict future states into a discrete time horizon using its internal *MPC*, NLP, and *Solver* modules, based on the cost function given by the *Objectives* constructed from the preplanned route and the online risk estimation (ORE) module, and applied control from the AMM system *AMMS*. The output of real-time computations from the ORE module as defined by the operators in a ROC is used both as a direct input to the MPC module during runtime, as well as serving as a measurement or monitoring tool for the ROC. Additionally, the AMMS is updated with events occurring in the environment detected by sensor measurements generated in the SAS, such as machinery failures.

The ANS module is in turn given a mission by the ROC, i.e., to follow a pre-planned route or path within certain time and risk limits, and uses the predicted trajectories to make autonomous control decisions and give commands to the AMMS. The resulting decisions, including the trajectory, risk cost calculations, and current internal states of the autonomous ship, are communicated to the ROC through information reports and electronic navigational charts (ENC) visualization, for supervisory human assessment and potentially human intervention.

Note that the mission given by the ROC to the ANS may be to follow a pre-planned route or mitigate damages through various emergency protocols. Moreover, the risk levels calculated by the ANS are of integral importance to the supervisory risk control, through scaling coefficients provided to the MPC objective or cost function for trajectory predictions. The risk analysis, as well as transforming the results of a qualitative analysis into a quantifiable optimization problem, thus provides the very important basis for the supervisory risk control and must be performed with care.

3.3 STPA Step 3: Unsafe supervisory control actions

Given the hazards presented in Table 5, hazardous or unsafe supervisory control actions (UCA) are to be identified with respect to the control flow in the system control architecture, as shown in Fig. 2. The supervisory control actions are given in Table 6, and the process variables considered during the identification of the control actions are given in Table 7. No command or SCA given by the ROC to the ANS is considered, as the ROC handles higher levels of decision-making than the ANS, and as such, is generally considered outside of the control bounds or complexity which the autonomous ship is expected to be able to control on its own.

Table 5 Specified hazards

H1a	Motion control objectives that result in violation of the minimum distance of separation to an obstacle are formulated, i.e., the cost function of the trajectory planning algorithm is inappropriately designed with respect to avoiding obstacles during assigned time intervals
H1b	Motion control objectives that do not result in violation of the minimum distance of separation to an obstacle are not followed, i.e., the subsystems of the ANS are unable to apply the required motion control to avoid obstacles
H2a	Motion control objectives that result in violation of the minimum distance of separation to the shore are formulated, i.e., the cost function of the trajectory planning algorithm is inappropriately designed with respect to avoiding the shoreline during assigned time intervals
H2b	Motion control objectives that do not result in violation of the minimum distance of separation to the shore are not followed, i.e., the subsystems of the ANS are unable to apply the required motion control to avoid the shoreline
H3a	Motion control objectives that result in sailing in too shallow water are formulated, i.e., the cost function of the trajectory planning algorithm is inappropriately designed with respect to keeping the vessel in deep enough waters during assigned time intervals
H3b	Motion control objectives that do not result in sailing in too shallow waters are not followed, i.e., the subsystems of the ANS are unable to apply the required motion control to keep the vessel within deep enough waters

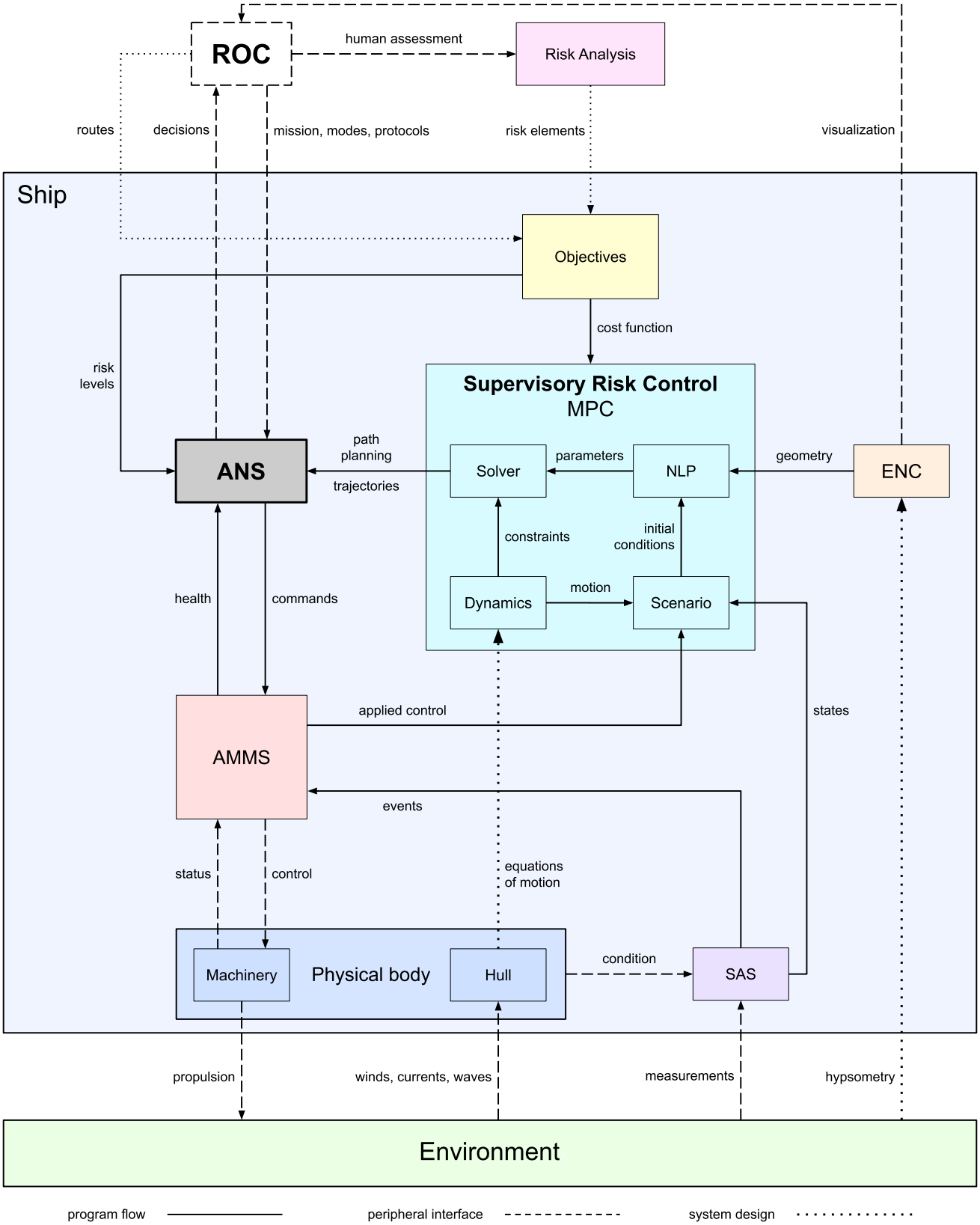

Fig. 2 Overview of the system control architecture. The supervisory risk control module is the primary scope of the design

Table 6 Considered supervisory control actions

Controller	Process	Supervisory control action
MPC	Solver	Calculate trajectory over time horizon
Solver	NLP	Solve NLP
NLP	ENC	Compute obstacle distances
Scenario	SAS	Construct scenario from states
MPC	Objectives	Compute risk costs
ANS	ORE	Assess risk levels
ANS	AMMS	Command first control step
AMMS	Machinery	Apply first control step

Table 7 Process variables

Process variable	Available states
Trajectory on horizon	Distinct future ship states
Current risk level	Continuous interval
Current wind disturbance	Constant velocity and angle
First MPC control step	Next optimal control input
Machinery health/status	Online/offline/fixed

It is assumed that the mission to be carried out by the ship is static throughout the decision process, and that no changes to objectives are made, i.e., the risk-aware cost function based on the results from the risk analysis and the specified mission objectives is unchanged during the entire process of autonomous navigation. This is assumed both in an effort to limit the scope of the system, and the fact that changing the mission to the ANS also is considered a higher level form of decision-making than the autonomous system should be able to perform by itself. Consequently, any change in the general mission objectives is thus considered to either be the initiation of an emergency protocol, or simply the start of another mission. For dynamic missions, the structure of the constructed solution would be unchanged, as the proposed analysis is generic. The same methodology may thus either be repeated in its entirety if a mission changes considerably, or one may alternatively change and re-evaluate the cost coefficients only (steps 12 and 13). It is suggested that an explicit methodology for online evaluation and tuning of the resulting cost function en route (for e.g., dynamic missions) should be investigated in future works.

The main SCA of the control hierarchy originates from commands given by the ANS to its two main modules: the MPC and the AMMS. Human operators in the ROC set the operational mode of the ANS to either autonomous mode or remote control, or initiate an emergency protocol during emergencies. If the ANS is in an autonomous mode, the MPC is run to predict the next optimal trajectory (i.e., the next set of future ship states within the sampled time horizon) by solving an NLP. While solving, the equations defining the ship dynamics are used to compute costs, based on calculated distances to spatial polygons of obstacles or grounding areas. The result is returned to the ANS as an optimal ship trajectory given the current ship and environmental states. It may be noted that trajectories are dependent on time, whereas paths (or routes) are not. Thus, the time aspect is in this context considered as one of the three main evaluation criteria during optimization, along with resource allocation and risk levels.

If the risk levels after the first control interval (or along the trajectory) are within accepted thresholds, the first control step of the trajectory is applied to the steering, power and propulsion machinery by the AMMS. Additionally, the current status and health of the physical machinery is estimated and reported back to the ROC for potential human intervention. The *Online* state means that both the steering and propulsion are available with ordinary capabilities, *Offline* indicates no propeller propulsion or steering available, and *Fixed* represents no available steering.

All identified UCA are listed in Tables 8 and 9, evaluated with respect to the ability of the ANS to keep the ship sufficiently safe during a specified time interval (i.e., the time required to apply appropriate SCA or counter-measures due to physical limitations or safety constraints). Note that complex computations, such as calculating future trajectories in this context are also considered supervisory control actions, due to the fact that they may *lead* to later (autonomous) decisions-making which can cause hazardous events. Each SCA is given a unique identifier and a short description, in addition to being assigned a relevant control action category label (mode) from Sect. 2.2.

Together, these UCA are used to identify the controller constraints (CC) presented in Table 10. From Fig. 2 and Table 8, it is clear that **UCA4–UCA12** are all part of subsystems related to computations for constructing scenarios and nonlinear problems to be solved by the MPC module. These UCA are thus closely related and connected, and are all found to potentially lead to the first three UCA identified between the ANS and the MPC. However, these underlying UCA also introduce separate system design concerns, which in this context are treated as supervisory control actions and must be considered during formulation of the individual controller constraints. The UCA presented in Table 9 are related to the ship in its current state during operations, i.e., either with respect to the maximum risk level threshold being violated or inadequate AMM or machinery control.

3.4 STPA Step 4: Safety constraints (SC) and loss scenarios

In the final step of the STPA, the system-level safety constraints of Table 4 and specified hazards of Table 5 are combined with the identified CC of Table 10 to identify

Table 8 Unsafe supervisory control actions (part 1)

ID	Controller	Process	Supervisory control action	Mode	Description
UCA1	MPC	Solver	Calculate trajectory on horizon	A	The calculated risk levels along the predicted trajectory is unacceptable, i.e., exceeds the risk threshold as defined by the ROC
UCA2	MPC	Solver	Calculate trajectory on horizon	B	The predicted trajectory returned by the MPC directly causes an obstacle allision or grounding
UCA3	MPC	Solver	Calculate trajectory on horizon	C/D	The MPC does not return a calculated trajectory within the required time interval, i.e., the result was provided too late or the computation was performed for too long
UCA4	Solver	NLP	Solve NLP	A	The computed solution trajectory is infeasible, i.e., a solution satisfying all physical constraints as well as risk constraints was not found, leading to **UCA1** or **UCA2**
UCA5	Solver	NLP	Solve NLP	B	The Solver produces a feasible trajectory which contains obstacle allisions or grounding events, leading to **UCA2**
UCA6	Solver	NLP	Solve NLP	C/D	A solution is not calculated within the required time interval e.g., due to a divergent or infeasible nonlinear problem, leading to **UCA3**
UCA7	NLP	ENC	Compute obstacle distances	A/B	The returned distances to obstacles are incorrect, producing an inaccurate or unsafe basis for the MPC trajectory calculations, leading to **UCA1** or **UCA2**
UCA8	NLP	ENC	Compute obstacle distances	C/D	The geometric operations or distance calculations applied to polygons are too computationally expensive, leading to **UCA3**
UCA9	Scenario	SAS	Construct scenario from states	A/B	The scenario classifications defined by the SAS are improperly formulated (e.g., if calm winds are classified as adverse weather), leading to **UCA1** or **UCA2**
UCA10	Scenario	SAS	Construct scenario from states	A/B	The scenarios produced are erroneous due to incorrect environment measurements or state estimations, leading to **UCA1** or **UCA2**
UCA11	MPC	Objectives	Compute risk costs	A/B	The calculated risk costs defined by the scaling coefficients from the ORE module are improperly formulated by the ROC, leading to **UCA1** or **UCA2**
UCA12	MPC	Objectives	Compute risk costs	C/D	The risk cost function is too computationally expensive, leading to **UCA3**

Table 9 Unsafe supervisory control actions (part 2)

ID	Controller	Process	SCA	Mode	Description
UCA13	ANS	ORE	Assess risk levels	A	The calculated risk level of the currently estimated ship position during a time interval exceeds the maximum risk threshold set by the ROC, leading to the activation of an emergency protocol, a change of control mode or adjustments made to mission objectives by the ROC
UCA14	ANS	AMMS	Command control step	A	The next ship position resulting from applying the first optimal control step calculated by the MPC exceeds the maximum risk threshold defined by the ROC, leading to **UCA13**
UCA15	ANS	AMMS	Command control step	B	The ship position resulting from applying the first optimal control step calculated by the MPC causes an obstacle allision or grounding
UCA16	AMMS	Machinery	Apply 1st control step	A/D	The control input as commanded by the ANS is not followed by the AMMS (i.e., the control is not applied to the machinery long enough or not at all), causing the maximum risk level threshold to be violated and leading to **UCA13**
UCA17	AMMS	Machinery	Apply 1st control step	B	The control input as commanded by the ANS is not followed by the AMMS, causing an obstacle allision or grounding
UCA18	AMMS	Machinery	Apply 1st control step	C/D	The AMMS or the ship machinery does not carry out the control commanded by the ANS within the required time interval e.g., due to physical system constraints or machinery faults

Table 10 Controller constraints

ID	UCA	Constraint description
CC1	1, 2, 3	The calculated trajectories on the horizon must be computed within the required time interval, cannot violate the maximum risk threshold, and shall not lead to any obstacle allision or grounding
CC2	4, 5, 6	A feasible NLP must be constructed, and the computed solution must converge to an optimal solution within the require time interval
CC3	7, 8	The calculations for obstacle distances must be sufficiently accurate as well as not too computationally expensive
CC4	9	All scenario definitions must be properly formulated, such that the behavior resulting from the calculated trajectories matches the expected behavior in any given scenario
CC5	10	The generated environment states from sensor measurements or simulations must be sufficiently accurate
CC6	11	The risk-based cost function and associated risk scaling coefficients must be correctly and sufficiently defined, such that obstacle allision or grounding does not occur due to logical or mathematical inconsistencies or assumptions that do not hold
CC7	13	Dependent on **CC6**, the maximum risk level threshold given by the ROC must be set with respect to the risk elements of the ORE module such that the threshold is violated only when the system should appropriately engage automatic emergency protocols or human intervention
CC8	14, 15	The first optimal control step of the MPC trajectory must not result in an obstacle allision or grounding
CC9	16, 17, 18	The physical machinery must carry out the control as commanded by the ANS through the AMMS within the required time interval

loss scenarios and UCA-level SC, ultimately presented in Table 11.

Note that **CC3**, **CC5** and **CC7** from Table 10 are disregarded, due to the following: The ENC and SAS modules are considered separate subsystems, which are assumed to independently perform adequately and within their given requirements. Similarly, setting the maximum risk threshold is in this paper considered outside the scope of the ANS, and a process that must be performed by human operators on the ROC. However, potential machinery faults (**CC9**) are included as part of the core of the supervisory risk control. The resulting safety constraints **SC1-5** are used as the fundamental basis for further decisions related to the design of relevant supervisory risk control components.

4 Methodology

The proposed steps for transforming the qualitative results from the STPA into a quantifiable optimal control problem for supervisory risk control are defined below. Details of the methodology are provided in the following subsections.

State variables identification

1. Define explicit mathematical state variables for all relevant measurable or quantifiable nouns or variables related to the identified loss scenarios and safety constraints of the risk-based supervisory control problem.
2. Represent the relationships between related variables as mathematical equations, and introduce intermediate variables or remove redundant variables if applicable.
3. Structure the identified variables into explicit system state and parameter vectors of the OCP.
4. Add all physical and logical equalities to the system dynamic equations of the OCP.

Safety inequalities construction

5. Formulate a risk-related inequality for all parts of a safety constraint that contains quantifiable variables.
6. Rank the safety inequalities based on the concept of risk priority numbers (RPN) [23].
7. Merge and/or remove any redundant safety inequalities by evaluation of assigned RPN and mathematical inspection.

Cost function formulation

8. Define a slack variable s for each inequality of the form $g(x) \geq 0$ such that $g(x) + s = 0$ holds, where x is a state variable.
9. Define an exponential cost term of the form $\mu e^{-\zeta s}$ for each slack variable s, where $\mu > 0$ and $\zeta > 0$ are tuning parameters approximately weighted according to the RPN assigned to its respective safety inequality.
10. Define the risk-related part of the cost function ρ as the sum of the exponential slack variable terms.
11. Formulate the total cost function as a sum of the resulting risk terms and other terms, e.g., related to resource consumption and mission objectives.

Evaluation and performance verification

12. Tune the coefficients of the cost function until the desired solver performance and control behavior is achieved.

Table 11 Loss scenarios and safety constraints

ID	Hazards	CC	Loss scenario	Safety constraint
SC1	**H1a, H2a, H3a**	**CC1**, **CC8**	The first control step or later intervals of the predicted future trajectory violates the minimum separation distance to an obstacle, the shore or too shallow water	The predicted ship trajectory must be spatially constrained to avoid crossing the minimum separation distance to obstacles, the shoreline or too shallow waters at all times
SC2	**H1b, H2b, H3b**	**CC1, CC2**	The MPC module is not able to compute a feasible ship trajectory within the required time interval, leading to a violation of the minimum separation distance to an obstacle, the shore or too shallow water due to inappropriate control during the computation period	Given some defined available computation power, the constraints of the NLP must be well-formulated and feasible to enable proper solver performance and satisfactory convergence rates within the required time interval, i.e., the ship model must sufficiently account for the ship dynamics, and the initial conditions and state constraints must be physically and logically consistent
SC3	**H1a, H2a, H3a**	**CC4**	The parameters or complexity of an estimated scenario for a given time interval are incorrect or insufficient, leading to unsuitable decision-making or MPC trajectory solutions which cause obstacle allision or grounding	The mathematical model of environmental variables and ship system states included in the risk-based cost function must be adequately formulated as to properly simulate the physical behavior of the ship in a designated scenario
SC4	**H1b, H2b, H3b**	**CC6**	The risk cost function of the NLP is too computationally expensive, which during computation leads to a violation of the minimum separation distance to an obstacle, the shore or too shallow water due to absent or sub-optimal decision-making by the ANS (inappropriate control)	The cost function must be computationally feasible with respect to the given computation power capabilities, within the specified time interval or calculation frequency, based on the dynamics of the ship. For instance, the cost function should be sufficiently smooth, locally convex, and computationally simple, such that the risk costs are easily calculated by the NLP solver
SC5	**H1b, H2b, H3b**	**CC9**	The machinery of the ship is not able to control the ship as required during the designated control interval, leading to drift-off or drive-off and a violation of the minimum separation distance to an obstacle, the shore or too shallow waters due to current ship velocities or external disturbances	The risk-based cost function must be designed and tuned such that the ship is sufficiently far enough away from grounding obstacles during normal operations as well as with limited propulsion allocation during unexpected failures or emergency scenarios, in order to avoid obstacle allision or grounding during drive-off or drift-off due to external disturbances. Namely, the risk levels of trajectories closer to minimum separation distances plus a safety distance margin must be weighted sufficiently high

13. Verify through inspection and test simulations that the performance satisfies all safety constraints.

This methodology leads to the development of a risk-based OCP and numerical solution. Note how the resulting safety inequalities and risk parts of the cost function are only related to the specific risk analysis performed through STPA, producing the mathematical *risk elements* of Fig. 2. Thus, all additional physical system constraints and resource consumption costs (i.e., fuel and/or time), as well as mission objective costs (e.g., path following) are in the steps 11 and 12 combined and tuned in tandem such that the total system is complete, meaning that the resulting cost function is appropriately weighted between the various aspects of autonomous navigation. This comparative tuning or weighting process through extensive testing and simulation, denoted as steps 12 and 13 of the methodology, must be performed in a case-by-case basis, and may be difficult to generalize. Even though risk analysis provides inputs to such a tuning, this procedure must be performed so that the formulation of the cost complete function is appropriate.

The following subsections apply each step of the proposed method for a case study presented in this work, and the performance of the implemented system is ultimately simulated and assessed in order to evaluate the quality of the mathematical formulation and risk quantification formulated by the procedure.

4.1 State variables identification

The physical state variables of the ship relevant to the control problem and their relationship equations are defined as given in the Appendix.

Next, the content of the SC description sentences are dissected and interpreted into additional mathematical variables by language analysis:

- **SC1**: Identified quantifiable nouns include the predicted ship trajectory, grounding obstacles, and the minimum separation distances to obstacles, the shoreline and too shallow waters.
- **SC2**: No quantifiable nouns or variables related to system states are identified in this safety constraint aside from the initial conditions of the NLP solver.
- **SC3**: No specific quantifiable nouns are identified aside from general mentions of physical states/behavior and (risk) cost function formulation.
- **SC4**: The only identified quantifiable noun is "risk cost".
- **SC5**: The identified quantifiable or measurable nouns are propulsion (allocation), risk level, (ship) trajectories, external disturbances, minimum separation distances, and a safety distance margin. Unexpected failures, drift-off, and drive-off are terms for special events or scenarios during extraordinary circumstances, and are consequently not measurable system states.

Some of the identified quantifiable nouns concern the same physical quantities: The predicted ship trajectory contains multiple ship states, which include the positions, orientations and velocities of the ship at each discrete time step within the given time horizon. These ship states, as well as propulsion forces, are defined in the Appendix. Moreover, the term *grounding obstacles* will throughout the remainder of this text encompass all possible allision obstacles, shorelines and/or too shallow waters. This simplification assumes that there are no consequence or outcome differences between ship grounding and obstacle or shoreline allision, as per the scope defined in Sect. 2.3.

The additional system state and environmental variables identified during Step 1 are listed collectively as follows:

- $\boldsymbol{X}_N = \begin{bmatrix} \boldsymbol{x}_0^\top & \dots & \boldsymbol{x}_N^\top \end{bmatrix}^\top$, the vector of N ship states $\boldsymbol{x}_k$ (including positions, orientations and velocities), i.e., the discretized ship trajectory throughout the predicted horizon of N control intervals
- $\sigma_j \in \Theta_J$ = one of J grounding obstacle polygons provided by the ENC (see Sect. 5.4)
- d_{sep} = the minimum allowed separation distance between the ship position $\boldsymbol{p}$ and any grounding obstacle σ_j
- d_{safe} = the safety distance margin which is added to d_{sep} in order to have the ship positioned sufficiently far away from obstacle boundaries such that temporary loss of propulsion does not result in grounding
- $\rho(\boldsymbol{x})$ = numerical value denoting the current risk cost of any ship state $\boldsymbol{x}_k$
- ρ_{max} = numerical threshold value denoting the maximum accepted risk (cost) level of the autonomous system at any point in the defined two-dimensional space, selected by the ROC with respect to the definition of ρ in the cost function
- v_d = velocity of the generalized disturbance forces acting on the ship during transit (i.e., wind velocity)
- ψ_d = angle of attack of the generalized disturbance forces acting on the ship during transit

All remaining system state and parameter vectors of physical constraints (such as propulsion or steering

limitations) not directly related to the risk analysis of the optimal control problem are given in Sect. 5.1.

4.2 Safety inequalities construction

The above mathematical definitions are subsequently used to construct safety inequalities (SI) for all sub-parts of each SC, given some appropriate interpretation of the SC formulations with respect to key logical, quantitative, qualitative and comparative statements.

In this work, only **SC1** and **SC5** are identified as containing safety inequalities. The remaining safety constraints are discussed in Sect. 4.4.

$$\textbf{SI1a} \quad \min\Big(d(\boldsymbol{x},\sigma_j)\ \forall\sigma_j \in \Theta_J\Big) > d_{\text{sep}}$$

$$\textbf{SI5a} \quad \min\Big(d(\boldsymbol{x},\sigma_j)\ \forall\sigma_j \in \Theta_J\Big) > d_{\text{safe}} + d_{\text{sep}}$$

$$\textbf{SI5b} \quad \min\Big(d(\boldsymbol{x},\sigma_j)\ \forall\sigma_j \in \Theta_J\Big) \cdot f(v_d,\psi_d) > d_{\text{sep}}$$

where $d(\boldsymbol{x},\sigma_j)$ is a distance function which returns the distance between its two arguments. The scaling function $f(.)$ dependent on the disturbance velocity v_d and disturbance angle of attack ψ_d is given as

$$f(v_d,\psi_d) = \max(0,\hat{\boldsymbol{i}}_j \cdot \hat{\boldsymbol{\omega}}) \cdot v_d \tag{1}$$

where $\hat{\boldsymbol{i}}_j$ is the unit vector from the ship to the obstacle j with the minimum distance to the ship position $\boldsymbol{x}$, and $\hat{\boldsymbol{\omega}}$ is the unit direction vector of the disturbance [13].

Next, each inequality is assigned a RPN [23], based on severity (result or consequence of failure/loss), occurrence (failure probability) and detection (failure identification difficulty). For the purpose of demonstration, three RPN are in this paper defined to serve as simple categories classifying the three identified safety inequalities: Recall that the loss scenario of **SI1a** is related to grounding without considering external disturbances, see Table 11. Table 12 shows the resulting RPN for each safety inequality, based on 1 to 10 rankings of its severity factor (S), probability of occurrence (O), and ease of detection (D) [24]. Note that the RPN of Table 12 are assigned with respect to how the *absence* of each cost term would affect the autonomous navigation behavior of the ship, e.g., a moderate probability of grounding is assumed if the short-term re-planning navigation algorithm does not include *any* term directly related to anti-grounding based on ENC.

Table 12 Process variables

Safety inequality	S	O	D	RPN
SI1a	9	4	3	108
SI5a	2	5	3	30
SI5b	9	6	4	216

In this example, an RPN of 108 is assigned to **SI1a** due to high severity with respect to grounding (9), a moderate probability of occurrence (4), and high ease of detection (3). This is considered appropriate due to the fact that the ship should preemptively follow a pre-planned feasible path within well-defined safety boundaries and parameters, with only dynamic and unplanned obstacles providing the main uncertainty aspect of the equation. If the MPC planner has to significantly re-plan the trajectory due to some unforeseen circumstances such as a crossing ship, the probability of grounding may consequently rise accordingly. Moreover, the onboard and remote sensors with respect to spatial movement in the environment are assumed to be relatively robust with sufficient levels of e.g., accuracy and redundancy shared across multiple different types of technologies, and should thus provide a reasonably high probability of detecting internal failures.

Next, **SI5a** is given an RPN of 30 due to low severity in which only the safety distance margin between the ship and the minimum distance of separation to obstacles is violated (2), moderate probability just slightly more likely than direct grounding (5), and similar failure detection capabilities (3). However, **SI5b** is given an RPN of 216 due to the following: This SI specifically is concerned with taking into account how external disturbances such as winds affect the ship trajectory, and how it relates to avoiding grounding events with respect to the physical propulsion and steering limits. Thus, the severity of grounding is high as in **SI1a** (9), the occurrence is somewhat higher due to expected disturbances (6), and the difficulty of detecting failures in the equipment for measuring disturbance forces is moderately low (4).

In general, the definition of RPN is closely related to risk acceptance of the system and its operation, which are usually determined based on stakeholders' perspectives, current risk levels for similar activities, rules, and regulations. Since risk acceptance is outside the scope of this paper, we have derived reasonable RPN for illustration. From this example, and with these assigned RPN, it is clear that there is a distinct disparity between the highly ranked **SI1a** and **SI5b** compared to **SI5a**. This will be addressed in the following steps.

4.3 Cost function formulation

All safety inequalities are subsequently transformed into risk cost terms which contribute to the accumulated system risk levels present at any point in time, and are used as additional guidance objectives during the autonomous decision-making and ship trajectory optimization.

In Step 8, the slack variables (SV) of each safety inequality are consequently defined as:

$$\begin{aligned} &\textbf{SV1a}: \quad s_1 = d_{\min}(\boldsymbol{x}) - d_{\text{sep}} \\ &\textbf{SV5a}: \quad s_2 = d_{\min}(\boldsymbol{x}) - d_{\text{safe}} - d_{\text{sep}} \\ &\textbf{SV5b}: \quad s_3 = d_{\min}(\boldsymbol{x}) \cdot f(v_d, \psi_d) - d_{\text{sep}} \end{aligned} \tag{2}$$

where $d_{\min}(\boldsymbol{x}) = \min\big(d(\boldsymbol{x}, \sigma_j)\big)\ \forall \sigma_j \in \Theta_J$. In general, $d_{\min}(\boldsymbol{x})$ is a non-smooth function, in which the J distances $d(\boldsymbol{x}, \sigma_j)$ to each grounding obstacle σ_j are themselves minimum distances between a singular geometric point $p(x, y) \in \boldsymbol{x}_k$ and the boundary of the obstacle polygon as provided by the ENC.

Note that the resulting SV, when feasible (greater than or equal to zero), indicate "increased" compliance with the SI constraint for larger values, directly analogous to the slack variables used by nonlinear numeric solvers to satisfy constraints through *barrier functions* [25].

The risk cost terms are during Step 9 constructed as monotonic and strictly increasing exponential functions with the (negatively) weighted SV as the exponents. Thus, the resulting risk cost function ρ for a single time interval k is by Steps 8 and 9 of the procedure defined as

$$\rho(\boldsymbol{x}_k, \sigma_j) = \mu_1 e^{-\zeta_1 s_1} + \mu_2 e^{-\zeta_2 s_2} + \mu_3 e^{-\zeta_3 s_3} \tag{3}$$

Note that the form of (3) directly follows from Step 9, and that the slack variables from Step 8 by definition serve as the only variables to be weighted or scaled through their respective coefficients as strictly positive cost terms. Moreover, (3) is only defined for a ship state during a single time interval ($\boldsymbol{x}_k$), with respect to an individual grounding obstacle polygon σ_j. It is proposed that this weighted cost may simply be summed for all grounding obstacles $\boldsymbol{\Theta}_J$ within the spatial horizon (see Sect. 5.3). Due to the exponential form of (3), far away obstacles are evaluated as negligible, making only nearby obstacles significant with respect to the total cost value at any point. This is indeed in accordance with the desired behavior, i.e., to first and foremost avoid nearby shorelines or shallow waters—due to the exponential function, any land mass consequently yields insignificant costs compared to, e.g., a small reef closer to the ship, if located behind it.

Determining the values of the risk coefficients $\mu_{1,2,3}$ and $\zeta_{1,2,3}$ is achieved approximately through the RPN of the SI associated with each resulting exponential term, i.e., larger coefficient values for higher RPN, and lower values for lower RPN. Consequently, smaller minimum distances between the location of the ship and grounding obstacles lead to (exponentially) larger costs, as expected. Moreover, each individual cost is weighted so that the terms with larger RPN have larger costs closer to their constraint boundary, with respect to the other terms. Note, however, that as the RPNs are semi-qualitatively defined, the correspondence between the RPN and the resulting cost scaling coefficients may after tuning be diminished.

More detailed discussion of the tuned risk cost terms with respect to optimal control and desired behavior for autonomous navigation is presented in Sect. 7.

4.4 Evaluation and performance verification

Evaluating and verifying the performance of the cost function and resulting NLP formulation are challenging. The coefficients of the cost function are tuned, e.g., incrementally through repeated simulations, and the performance is evaluated by comparing the resulting behavior with all safety constraints. Section 4.2 denotes how **SC1** and **SC5** are exercised through **SI1a**, **SI5a** and **SI5b**. However, **SC2**, **SC3** and **SC4** are directly related to the solver performance, i.e. that the resulting solution convergence and behavior is appropriate and achieved within the required time intervals. Compliance with these constraints are thus verified through the following discussion and simulation results presented in Sect. 6, both validating the proposed methodology in this work as well as the autonomous behavior resulting from the application of the method to the considered use case. A special case worth noting is nevertheless that the resulting cost function, as defined by following the proposed methodology, by construction satisfies **SC4** with respect to computational feasibility.

The initial solution given to the solver (i.e., the pre-planned trajectory) during the first solve will greatly affect the produced trajectories due to the problem being non-convex, and must be pre-computed appropriately. The solver used to calculate optimal trajectories at each time interval may also have a significant impact on the generated results. In this work, a gradient-based solver is used—non-smooth functions such as the minimum function used for polygon distances are handled by the solver through automatic differentiation (differentiable programming). However, one may consider using solvers employing e.g. evolutionary algorithms, particle swarm optimization (PSO) or similar techniques, in order to utilize discontinuous and non-differentiable cost functions.

Additionally, the non-convexity of the constructed NLP makes no guarantees with respect to optimality of its local solutions, and the generated ship trajectories must consequently be assessed during development, with respect to the current conceptual view of acceptable risks, desired operations behavior and defined mission objectives. As such, the implementation and performance of a MPC algorithm and a case study with example simulations are presented in the following sections for evaluation and verification purposes.

5 NLP and MPC formulation

This section presents the resulting MPC and final cost function to be solved by the NLP algorithm, extended and improved from previous works [13]. Additionally, simulation results of the various test scenarios are summarized to demonstrate the performance of the formulated risk-based supervisory control problem.

5.1 Optimal control problem (OCP) formulation

An OCP is in general defined as follows:

$$\begin{aligned} \min_{\boldsymbol{x}(.),\boldsymbol{u}(.)} & \int_{t=0}^{T} \tilde{\phi}(\boldsymbol{x}(t),\boldsymbol{u}(t),\boldsymbol{\theta}(t))\mathrm{d}t \\ \text{s.t.} \quad & \dot{\boldsymbol{x}}(t) = \boldsymbol{f}(\boldsymbol{x}(t),\boldsymbol{u}(t),\boldsymbol{d}(t)) \\ & \boldsymbol{h}(\boldsymbol{x}(t),\boldsymbol{u}(t)) \leq \boldsymbol{0} \\ & \boldsymbol{x}(0) = \boldsymbol{x}_0, \quad 0 \leq t \leq T \end{aligned} \tag{4}$$

where $\tilde{\phi}$ is a scalar stage cost function that will be defined in Sect. 5.3, $\boldsymbol{\theta}$ is a parameter vector, $\boldsymbol{x}_0$ is the initial state, T is the prediction horizon, and $\dot{\boldsymbol{x}}$ is given by the ship dynamics as presented in the Appendix. The hard constraints $\boldsymbol{h}(\boldsymbol{x}(t),\boldsymbol{u}(t))$ are given as:

$$\begin{aligned} -f_{\max} &\leq u_1 \leq f_{\max} \\ -\omega_{\max} &\leq u_2 \leq \omega_{\max} \end{aligned} \tag{5}$$

where $\boldsymbol{u} = \begin{bmatrix} u_1 & u_2 \end{bmatrix}^\top$ is the control input vector where u_1 is the propulsion force of the rudder, u_2 is the rotational turning rate of the rudder, and $f_{\max}$ and $\omega_{\max}$ are the maximum propulsion force and rotational turning rate of the rudder, respectively. The solution to problem (4) will be deployed in a receding horizon fashion, yielding an MPC scheme.

5.2 The reference path

A preplanned reference path is defined as a piecewise linear (spline) function given an initial position, discrete intermediate points and a destination. Next, the reference path is parameterized, giving the two-dimensional reference function

$$\boldsymbol{r}(\alpha) = \begin{bmatrix} x(\alpha) \\ y(\alpha) \end{bmatrix} \tag{6}$$

for calculating path points where $\alpha \geq 0$ is a scalar advancement parameter (i.e. the traveled distance during a control interval) acting as a decision variable along the preplanned path, and $x(\alpha)$ and $y(\alpha)$ are piecewise linear functions.

5.3 Objectives and cost function definitions

In order to construct the NLP, a cost function to be minimized is defined. Advancing along the path is a simple matter of increasing α, and the desired ship speed along the path is furthermore established by penalizing ship transit velocities larger than the given reference speed s_{ref}. This is achieved by minimizing a speed penalty decision variable β, where

$$u^2 + v^2 \leq s_{\text{ref}}^2 + \beta, \quad 0 \leq \beta \tag{7}$$

in which u is the forward surge velocity, and v is the sideways sway velocity.

By collecting the additional decision variables into a vector for each time step through the control horizon N, we have

$$\boldsymbol{q}_k = \begin{bmatrix} \alpha_k \\ \beta_k \end{bmatrix}, \qquad k = 0, 1, \ldots, N-1 \tag{8}$$

which in Sect. 5.4 is used to define the NLP decision variable vector $\boldsymbol{w}$.

In this work, the cost function is defined with the purpose of producing a safe ship trajectory that fulfills the mission objectives:

$$\phi(\boldsymbol{w},\boldsymbol{\theta}) = \sum_{k=1}^{N} \xi(\boldsymbol{x}_k,\boldsymbol{q}_k,\boldsymbol{q}_{k-1}) + \epsilon(\boldsymbol{u}_k,\boldsymbol{u}_{k-1}) + \rho(\boldsymbol{x}_k,\boldsymbol{\theta}_k) \tag{9}$$

The cost terms are defined as follows:

(i) The path progression cost function

$$\xi(\boldsymbol{x}_k,\boldsymbol{q}_k,\boldsymbol{q}_{k-1}) = \boldsymbol{\kappa}^\top \begin{bmatrix} ||\boldsymbol{r}(\alpha_k) - \boldsymbol{p}_k||^2 \\ ||\alpha_k - \alpha_{k-1} - \alpha_{\text{trav}}||^2 \\ \beta_k \end{bmatrix} \tag{10}$$

where $\boldsymbol{\kappa} > \boldsymbol{0}$ is a vector of tuning parameters. These terms are responsible for advancing the ship position $\boldsymbol{p}_k$ (trajectory) along the precomputed feasible path, through the constant path step parameter α_{trav} and the reference path $\boldsymbol{r}(\alpha_k)$. The β_k term penalizes violations of the transit speed reference as detailed in Sect. 5.3. It is recommended that α_{trav} is chosen such that $s_{\text{ref}} \approx \alpha_{\text{trav}}/t_\Delta$, where t_Δ is the sampling period of the NLP.

(ii) Next, the control input cost function is defined as

$$\epsilon(\boldsymbol{u}_k,\boldsymbol{u}_{k-1}) = \boldsymbol{u}_k^\top \boldsymbol{\Lambda} \boldsymbol{u}_k + (\boldsymbol{u}_k - \boldsymbol{u}_{k-1})^\top \boldsymbol{\Delta} (\boldsymbol{u}_k - \boldsymbol{u}_{k-1}) \tag{11}$$

where $\boldsymbol{\Lambda} = \text{diag}(\lambda) > \boldsymbol{0}$ and $\boldsymbol{\Delta} = \text{diag}(\boldsymbol{\delta}) > \boldsymbol{0}$ are tuning matrices. These terms collectively help conserve power and reduce the input variations, consequently lowering environmental and operational costs.

(iii) Finally, the constructed risk cost function is used to keep the grounding risk levels low:

$$\rho(\boldsymbol{x}_k, \boldsymbol{\theta}_k) = \sum_{j=1}^{J} \mu_1 e^{-\zeta_1 s_{1,j}} + \mu_2 e^{-\zeta_2 s_{2,j}} + \mu_3 e^{-\zeta_3 s_{3,j}} \quad (12)$$

where $s_{1,j}$, $s_{2,j}$ and $s_{3,j}$ are defined as in (2), with respect to each grounding obstacle $\sigma_j \in \boldsymbol{\Theta}_J$. Here, the dot product within $s_{3,j}$ from (1) scales the disturbance contribution toward the grounding obstacles in any orientation around the ship, i.e., increasing the risk close to an obstacle to the east of the ship if the wind, waves or currents are coming from the west, etc. Negative dot products are however set to zero, disregarding "favorable" disturbances with respect to perceived risks. The remaining variables were defined in Sect. 4.3.

It may be noted that all of the initial safety inequalities are transformed into risk costs, in favor of being formulated as explicit constraints to ensure safe distances between the ship and obstacles. Thus, this "soft constraint" formulation utilizes violatable risk costs in order to acknowledge that grounding risks may still be evaluated even if high, and guaranteeing NLP feasibility. Using exponential terms for the obstacle or grounding risk costs serves to strongly dominate the other objectives in the cost function, heavily favoring staying safe from grounding obstacles. The grounding risk sensitivity constant ζ may for this purpose be tuned for optimal behavior.

5.4 Nonlinear programming

The dynamic ship model is discretized in order to solve the problem numerically. The continuous time variable t is divided into a time grid of N intervals on the horizon T, defined by discrete time instants $t_k \in \{t_0, t_1, \ldots, t_N\}$. The system inputs are discretized as piecewise constant over that time grid, i.e., $\boldsymbol{u}_k = \boldsymbol{u}([t_k, t_{k+1}])$. The system state is discretized using a numerical integration function $\boldsymbol{x}_{k+1} = \boldsymbol{F}_k(\boldsymbol{x}_k, \boldsymbol{u}_k, \boldsymbol{d}_k)$, based on the widely used *Runge–Kutta 4th order method*. The discretization allows one to treat (4) as a nonlinear program by defining a vector of decision variables

$$\boldsymbol{w} = \begin{bmatrix} \boldsymbol{x}_0^\top & \boldsymbol{q}_0^\top & \boldsymbol{u}_0^\top & \ldots & \boldsymbol{x}_{N-1}^\top & \boldsymbol{q}_{N-1}^\top & \boldsymbol{u}_{N-1}^\top & \boldsymbol{x}_N^\top & \boldsymbol{q}_N^\top \end{bmatrix}^\top \quad (13)$$

where $\boldsymbol{q}_k$ is the vector of additional decision variables related to mission objectives defined in Sect. 5.3. Additionally, a parameter vector comprised of various control settings, desired states and coefficients through time is denoted as $\boldsymbol{\theta} = \begin{bmatrix} \boldsymbol{\theta}_0^\top & \ldots & \boldsymbol{\theta}_N^\top \end{bmatrix}^\top$. The parameters considered in this example are

$$\boldsymbol{\theta}_k = \begin{bmatrix} s_{\text{ref}} \\ \alpha_{\text{trav}} \\ \boldsymbol{\Theta}_J \end{bmatrix} \quad (14)$$

for all t_k where s_{ref} is a constant reference transit speed and α_{trav} is a path progression parameter (see Sect. 5.3) across all N control intervals. The grounding obstacles are in this work modeled as a union of convex polygons $\sigma_j \in \boldsymbol{\Theta}_J$. Thus, $\boldsymbol{\Theta}_J$ is the collection of all grounding obstacles, enumerated by $j = 1, \ldots, J$ in (12).

In other words, the entirety of all grounding or allision polygons present in some specified data (sub)set provided by the ENC module is included for distance calculations in each time step, yielding J distance computations for each subsequent NLP solver iteration, for each time interval k throughout the time horizon of N sampling intervals. Moreover, the distance to each polygon is computed based on its inherent complexity, given by its number of vertices and edges. Thus, it is important to limit the resolution of the grounding polygons such that the computation time is fast enough (**SC4**), e.g., by simplifying and reducing the number of vertices of each ENC polygon to some extent relative to the size of the ship or the size of the considered environment during application runtime. Note, however, that simplifying the complexity, which consequently also lowers the resolution of the grounding obstacles, directly affects the spatial error margins needed for safe navigation near the obstacle polygons. As a result, both d_{sep} and d_{safe} from (2) must be defined with respect to the resolution and inherent complexity of the grounding obstacles as provided by the ENC module (**SC3**).

The resulting NLP is defined as

$$\begin{aligned} C(\boldsymbol{\theta}, \boldsymbol{X}_0) = \min_{\boldsymbol{w}} \quad & \phi(\boldsymbol{w}, \boldsymbol{\theta}) \\ \text{s.t.} \quad & \boldsymbol{g}(\boldsymbol{w}) = \boldsymbol{0} \\ & \boldsymbol{h}(\boldsymbol{w}) \leq \boldsymbol{0} \end{aligned} \quad (15)$$

where $C(\boldsymbol{\theta}, \boldsymbol{X}_0) \in \mathbb{R}$ is the minimum cost generated by a given set of parameter values and initial conditions $\boldsymbol{X}_0$, i.e., a full initial trajectory of N ship state vectors $\boldsymbol{x}$ given by the path $\boldsymbol{r}$: The preplanned path and ship speed reference parameters are used to calculate a reasonable initial guess for the ship trajectory $\boldsymbol{X}_N$. For all subsequent NLP solve calls, the last solution (forward shifted one time step) is used as the initial guess, i.e. warm start. The inequality constraints $\boldsymbol{h}(\boldsymbol{w})$ are given by (5), and the equality constraints $\boldsymbol{g}(\boldsymbol{w})$ hold the system dynamics:

$$\boldsymbol{F}_k(\boldsymbol{x}_k, \boldsymbol{u}_k, \boldsymbol{d}_k) - \boldsymbol{x}_{k+1} = \boldsymbol{0}, \quad k = 0, 1, \ldots, N-1 \quad (16)$$

The cost function ϕ was defined and discussed in Sect. 5.3, based on the risk cost formulation of Sect. 4.3. Note that the discretization chosen here is based on the *direct*

multiple-shooting approach [26]. Because of the nonlinear dynamics and since the obstacles yield a non-convex feasible set, the NLP (15) is non-convex. As a result, the goal is to compute a feasible and *local* optimal solution for a given control horizon N and initial conditions. Moreover, only cost balancing is noted utilized to achieve the desired control behavior, rather than using hard constraints in addition to the ship dynamics and the natural input constraints. This ensures feasibility of the NLP solutions.

6 Results

6.1 Simulation verification

The purpose of this section is to present simulation results which showcases how the constructed risk cost terms affect the performance and resulting trajectory of the MPC scheme, to serve as the verification method for this work. Thus, the goal is to verify that the safety constraints of Table 11 are satisfied during the autonomous navigation shown in the simulations, with respect to suitable compliance relative to the expected behavior. The model parameters used for simulation verification are presented in Table 13.

Figure 3 presents a reference trajectory resulting from a simulation using the MPC scheme discussed in this paper is presented in, for the purpose of comparison to later simulation trajectories. A square example region with an area of 9 km^2 just south of Giske island in Norway is chosen to demonstrate various aspects and discussion points related to the construction of the risk cost terms. Using the visualization capabilities of the ENC package [22], several information

Table 13 Model parameters

Parameter	Symbol	Value	Unit
Path step size	α_{trav}	75	m
Overall ship length	L_{oa}	75	m
Transit reference speed	s_{ref}	2.5	m/s
Frontal projected wind area	A_{Fw}	110	m^2
Lateral projected wind area	A_{Lw}	624	m^2
Max propulsion force	$f_{\max}$	400	kN
Max rudder turn rate	$\omega_{\max}$	2.0	rpm
Viscous damping force surge	X_u	$5.0 \times 10^1 1$	kNs/m
Viscous damping force sway	Y_v	2.0×10^2	kNs/m
Viscous damping force yaw	N_r	3.0×10^4	kNs/rad
Hydrodynamic added mass surge	$X_{\dot{u}}$	4.4×10^4	kg
Hydrodynamic added mass sway	$Y_{\dot{v}}$	8.6×10^5	kg
Hydrodynamic added mass yaw	$N_{\dot{r}}$	4.9×10^7	kgm^2
Rotational inertia yaw	I_z	9.8×10^8	kgm^2
Rigid-body ship mass	m	1.5×10^6	kg

Fig. 3 A demonstrative path following simulation using MPC, not including any risk cost terms in the cost function

overlays are added. Three waypoints are given as a route reference (the western-most off-screen), and a green path is drawn between the waypoints. The ship trajectory is shown as a white trail of ship poses (i.e., the position and orientation of the ship at each time interval), and the future trajectory solution computed by the MPC is shown in yellow. The current ship pose is shown in magenta, recorded as a snapshot during a simulation run. It may be noted that the path following and resource consumption costs are tuned such that the trajectory is allowed to deviate slightly from the planned path, in order to save time, rudder actuation, or fuel into the time horizon.

Figure 4 shows a simplified example in which only the first risk cost term for **SV1a** is included in the complete cost function during the MPC run. Moreover, only one single island is considered as a grounding obstacle in this example, for clarity. Shown in red, the single grounding obstacle is constructed as the convex hull of all ocean depths more *shallow* than 10 m, closest to the initial ship position. The first waypoint serves as the starting point of the simulation, and is shown as a green disk. The convex hull of the original grounding obstacle is chosen, based on the assumption that concave crevices along the boundary of any obstacle polygon are not considered for purposeful navigation along a planned path. Furthermore, though the island itself (i.e., land mass above sea level) is not intersected by the planned path, the convex grounding obstacle intersects slightly. This is as a result of an additional safety margin added to the grounding obstacle in all directions, in Fig. 4 set to 50 m. This static margin is added at the ENC level in order to ensure

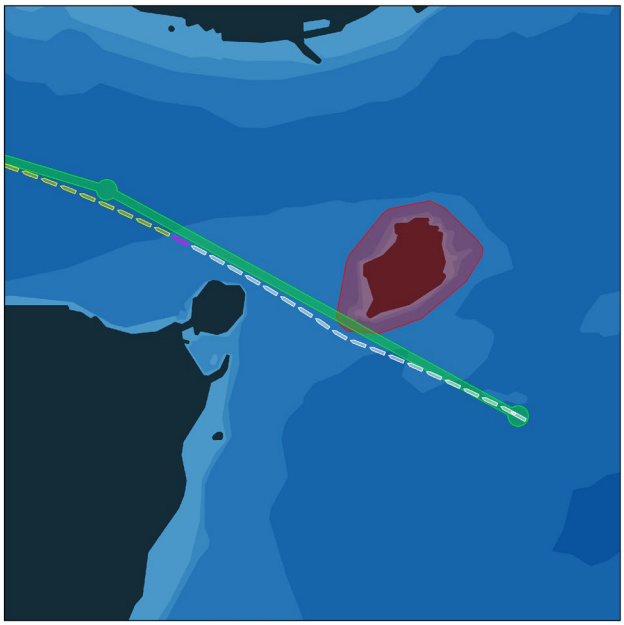

Fig. 4 MPC simulation showcasing the effect of the first risk cost term (**SV1a**) for only one grounding obstacle shown in red

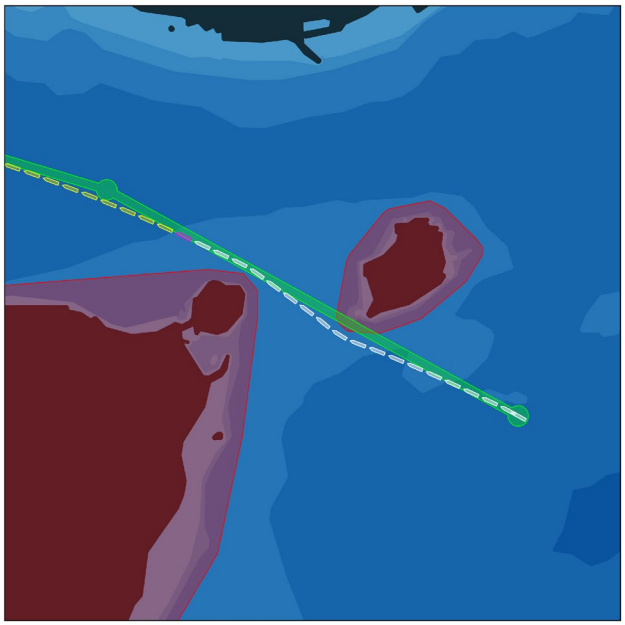

Fig. 5 MPC simulation utilizing the first risk cost term (**SV1a**) to avoid two opposing islands at each side of the ship

that inaccuracies related to the numerical charts data and/or e.g., tides are accounted for, and is considered a separate discussion than the cost term added related to the d_{safe} variable of **SV5a**. This static margin should be kept small, and is exaggerated in this work for the purpose of demonstration. See the discussion related to Fig. 9 on this topic.

The consequence of the combination of these factors is that the resulting optimal trajectory computed by the MPC module deviates from the planned path close to the grounding obstacle, as expected. Thus, it is apparent that the first risk cost term is sufficient to produce the necessary behavior in order to avoid grounding obstacles, even when the grounding obstacle is intersecting with the original path.

The next example is shown in Fig. 5, which includes the south-western island group as a grounding obstacle. Notice the small perturbation of the ship trajectory close to the northern tip of the obstacle, indicating that both islands have an effect on the risk cost term, as expected. Moreover, it is shown that the risk cost is properly defined also for grounding obstacles located on opposite sides of the ship. The northern island is, however, located far away, and is through the inverse exponential weighting and resulting value of the risk cost term negligible compared to the closer islands. This is considered appropriate for the specific environment domain shown in the example demonstrations in this work.

Figure 6 demonstrates how the addition of the second risk cost term related to **SV5a** appropriately adds an extra virtual safety buffer or safeguard with respect to nearby grounding obstacles, indeed in accordance with the wording of **SC5**: The risk levels of trajectories closer to the minimum separation distance plus a safety distance margin must be weighted sufficiently high, such that the ship to a larger degree is able to avoid grounding if unexpected failures or disturbances are introduced—which effectively increases the time available to, e.g., restart the ship engines after a power blackout in order to regain ship control. The trajectory

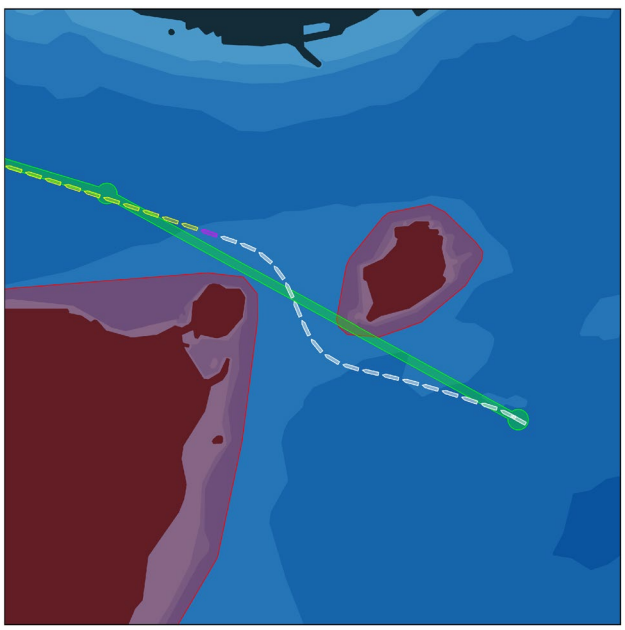

Fig. 6 MPC simulation with increased grounding sensitivity through the second risk cost term (**SV5a**), or alternatively by tuning of the ζ coefficient

shown in Fig. 6 is ultimately considered an appropriate trajectory for sufficiently safe navigation through the strait as presented here.

It may be noted, however, that this behavior also can be achieved in this example simply by adjusting the sensitivity coefficient ζ_1 of the first risk cost term, as may be inferred through the mathematical similarities of both terms. The second risk cost term is nevertheless included in this work, for completeness.

In contrast to the previous example demonstrations, Fig. 7 shows a simulation in which the non-convex forms of each grounding obstacle are used in place of the previous convex hulls. Additionally, the minimum distances between the ship and both nearby grounding obstacles are shown as orange lines, in order to clearly visualize the exact coordinates of each distance calculation provided to the exponential risk cost terms during MPC optimization.

Note, however, that the resulting trajectory of Fig. 7 is identical to that of Fig. 6. This is a consequence of the fact that only the minimum distances between the ship and all obstacles are provided to the risk cost terms. By definition, no point not located exactly on the boundary of the convex hull of an obstacle may exist as a minimum distance to the ship. Thus, the advantage of calculating the convex hulls of all obstacles during initialization of the algorithm through the functionality provided by the ENC module is clear. This results in fewer polygon vertices for distance computations, which leads to faster solver performance in accordance with the objective described in **SC4** related to computational feasibility. The approach may nevertheless also be used on non-convex obstacles if necessary, at a price of reduced computational efficiency.

Figure 8 presents the same orange distance visualizations as those of Fig. 7, applied to the convex grounding obstacles in Fig. 6. In addition to the previous discussion with respect to convexity, it is apparent that these line segments are more well-behaved and the variation between each consecutive line is smaller, which in turn improves the smoothness of the gradients as present in the MPC during the NLP solve. This effect is seen directly by faster solver timings.

The last example related to the first two risk cost terms and the structure of the grounding obstacles provided by the ENC is presented in Fig. 9, in which the safety buffer added to the obstacles are increased from 50 to 200 m. This is shown in order to demonstrate the effects of this static approach compared to the addition of the increased safety risk cost applied through the second risk cost term.

One may notice how the resulting trajectory in this case is significantly more restricted through the consequently narrower isle strait. It may be argued that the approach of adding static safety margins in this manner—i.e., through the initial creation process and structure of grounding obstacles, as constructed by the ENC module—reduces the flexibility of the MPC algorithm, and should be avoided in favor of the added virtual safety margin cost exemplified through the second risk cost term in this work. Moreover, it is clear that if the safety buffer is too large (e.g., 400 m),

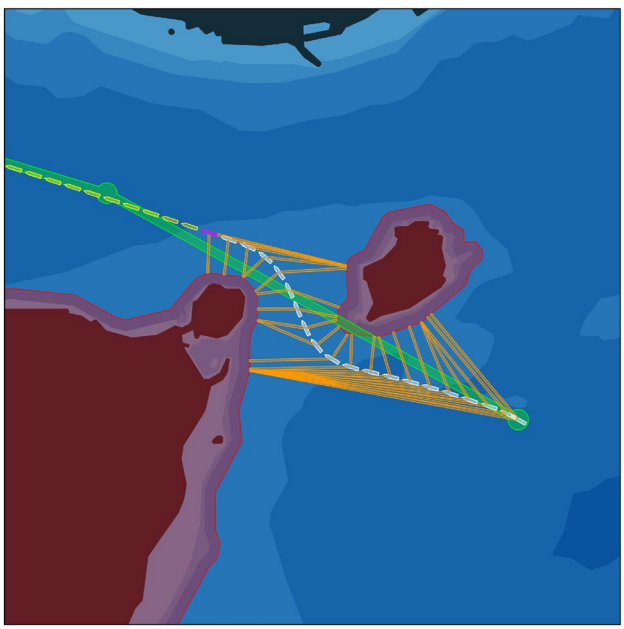

Fig. 7 MPC simulation with non-convex grounding obstacles, and minimum distances shown as orange lines between each past vessel position and the obstacle polygons

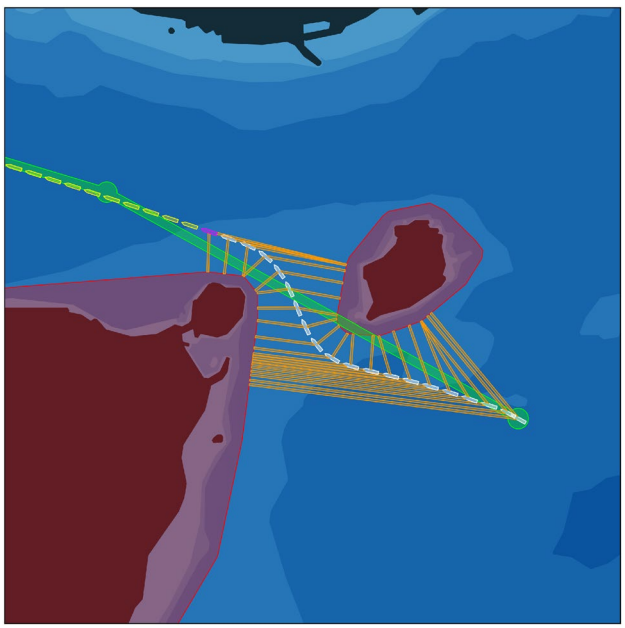

Fig. 8 MPC simulation showing the orange minimum polygon distances for convex grounding obstacles. Less variance between each evaluated minimum distance along the trajectory yields improved solver performance

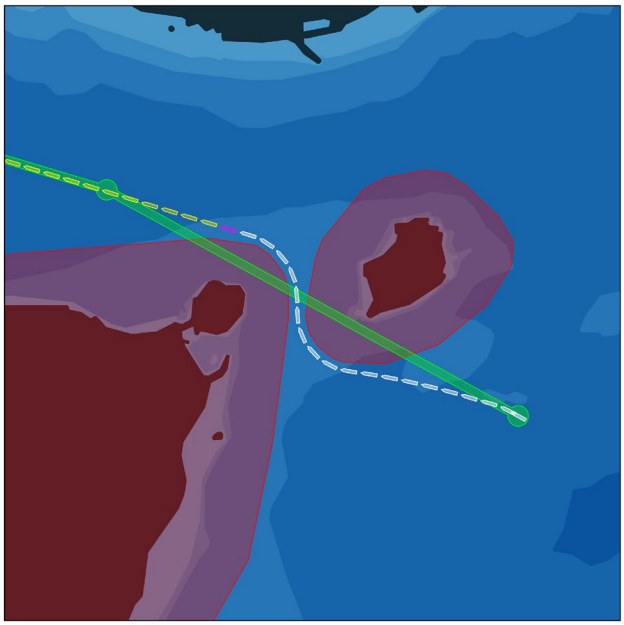

Fig. 9 MPC simulation with an increased safety buffer added to the convex obstacle polygons, resulting in less flexible performance and lower degrees of solver feasibility

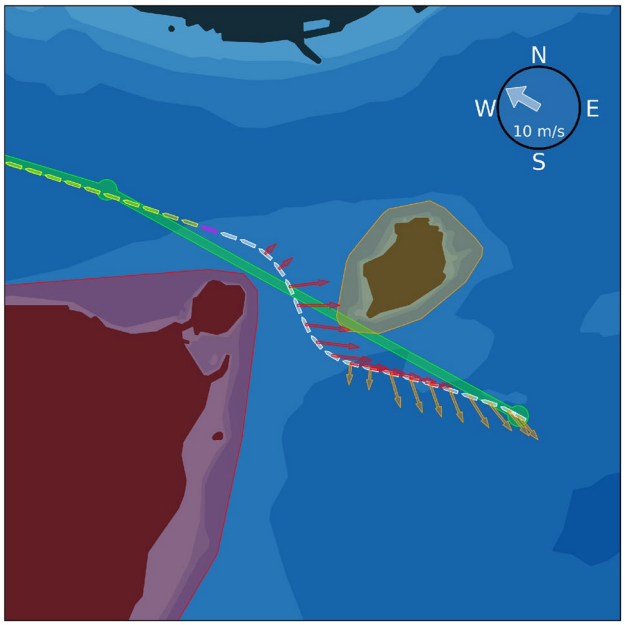

Fig. 10 MPC simulation including visualization of the third risk cost term (**SV5b**), demonstrating the effects of a wind disturbance with respect to nearby grounding obstacles

the strait would be entirely closed, which would lead to an entirely new behavior in which the ship must sail around the resulting merged obstacle of both islands. Though potentially appropriate in some cases, this approach is considered unstable and prone to produce irregular solutions given any particular environment.

The effects of the third and last risk cost term of the constructed risk cost function produced through the methodology presented in this paper is visualized in Fig. 10. Though the effects of the cost term is limited with respect to changes in the trajectory due to the direction of the wind compared to the vessel heading, the example illustrates how both isles each contribute separately to the risk cost scaling. Here, the two grounding obstacles located at the port and starboard side of the ship are given the colors red and orange, respectively. Furthermore, the value of s_3 at each time step is multiplied by the unit vectors with directions equal to the *opposite* of the direction of the vectors between the ship center and the minimum distance point on each obstacle. The wind disturbance direction and wind velocity is shown in the compass in the top-right corner of the environment plot.

Intuitively, the red and orange arrows point away from their respective grounding obstacles at each side of the ship, due to the risks increasing *toward* the obstacles—which the MPC attempts to minimize, effectively directing the ship *away* from the obstacles in accordance with the arrow directions. Moreover, the length of the arrows are proportional to the gradient of the risk cost term at each point, i.e., the magnitude of how much the perceived risks promote evasive maneuvers with respect to each grounding obstacle.

This illustration aims to visualize how the external disturbances (here limited to wind disturbances only) affect the risk levels through the third risk cost term. In this case, the wind direction is in an on-land direction toward both grounding obstacles with respect to the initial location of the ship, yielding positive scalar products. Thus, if the wind force is driving the ship toward the shoreline or other grounding obstacles, the risk level increases as expected.

Figure 11 presents a situation in which the wind direction is directed toward the southwest grounding obstacle shown in red. Notice how the trajectory in this case intersects the orange grounding obstacle, due to the scalar products between this obstacle and the ship location being non-positive and consequently disregarded. This is indeed the expected behavior, as this risk cost term is only concerned with the risk levels related to external disturbance forces, and must be combined with the first (two) risk cost term(s) in order to produce appropriate trajectories during autonomous navigation. It is argued that no disturbance forces should be included as a positive or favorable driving force toward safe autonomous control, and as such is factored out in this context.

Lastly, the quality of the solutions is considered appropriate: All figures are generated within 6.5–12.4 s on an Intel® Core™ i7-9700K (3.6 GHz), with a 20 min future horizon using a sampling time of 30 s. This means that the algorithm is able to predict, optimize and plot the future ship

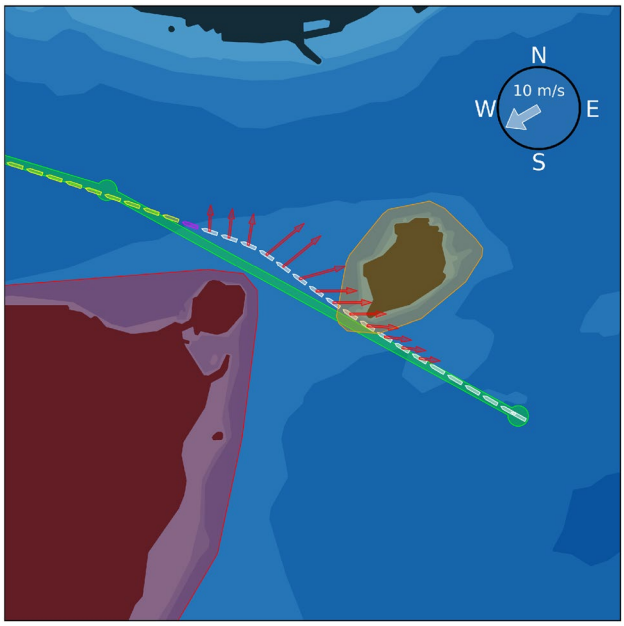

Fig. 11 MPC simulation visualizing how the third risk cost term (**SV5b**) is only significant for on-land wind directions

states along the route for 40 time intervals into the future, repeated 20 times (one for each trailing ship pose shown in gray), within a maximum of 12.4 s on a desktop computer. The average optimization timing for a single run is thus well below one second. Consequently, the proposed algorithm may be repeatedly utilized online to predict the real-time effects of the latest measured or predicted weather conditions on a voyage 20 min into the future, recalculated every second using this setup. It is however noted that this performance is strongly dependent on the trade-off between the chosen data resolution and the resulting solution quality. Moreover, the MPC solutions consistently convergence to similar sets of trajectories across various tested simulations of different initial conditions, further assuring the validity of the approach. This concludes the verification of the constructed risk cost terms, by human assessment and review of the presented simulation demonstrations.

7 Discussion

7.1 Choice of methods: STPA and MPC

The risk analysis part of this work is based on STPA, mainly due to its feasibility for large and complex control system structures such as autonomous navigation, control and awareness systems of a ship presented in this work. By focusing on potential unsafe control actions, loss scenarios, and associated safety constraints, the integrity and safety of the system is thus considered through the emergent behavior of the interconnected system as a whole.

For the quantifiable and optimization side of the problem, a MPC scheme was chosen due to the flexibility and robustness of the method. As presented and discussed in this paper, the MPC approach is largely capable of solving quite complex optimization problems if given appropriate and well-defined system dynamics and cost formulations. In this regard, the difficult part of the method is rather to provide the MPC with a feasible and satisfactory cost function, as well as an initial guess that produces the desired results when used for autonomous navigation.

7.2 Risk analysis vs optimal control

There is a definite distinction between the fields of qualitative risk analysis and numerical optimal control. As such, it is challenging to standardize a transformation between STPA and MPC. First, STPA is considered an effective method for identification of hazardous events for a range of applications. However, the results of the analysis often yield extensive collections of possible failures or unsafe control actions. A challenge with STPA is that it only considers negative losses, which means that any rewards and trade-offs between risks and system performances are not analyzed or supported in decision-making. Furthermore, STPA is a qualitative approach, which means that several additional steps are needed to translate the results into meaningful representations in MPC. A methodology for such a "translation" is provided in this paper.

One of the main strengths of the traditional applications of MPC is that it usually has relatively straightforward and well-tested costs, similar to feedback controllers, such as the linear-quadratic regulator (LQR). If instead the cost terms of the MPC are extensively non-smooth or nonlinear, feasibility and solvability problems for a given real-time constraint may arise. It is also apparent that operational optimization is not the same as emergency management. During extreme conditions, the focus is arguably only to handle or contain the situation to a satisfactory degree, to avoid further loss of control within strict and short time periods. Hence, there is usually limited benefits to be gained from optimizing the best possible solution during these scenarios. The method is nonetheless chosen due to the parallels between MPC and human decision-making, in the sense that humans inherently weigh costs (negative consequences) and rewards against each other when making most logical decisions on a day to day basis. The challenge, as previously mentioned, lies in the uncertainty and intricacies that arise when quantifying the decision variables for optimization, and as such, this must be performed with care.

7.3 risk cost verification

There is no conclusive way to verify if the constructed costs of the OCP and NLP are sufficient to satisfy the given safety constraints for *any* possible scenario. Generally, some form of evaluation method has to exist for any given method for which the performance is to be examined. However, for the performance to be evaluated and classified, real meaning and/or actual values need to be assigned to abstract concepts for the purpose of assessment, as it is not possible to know the "true" or objective risk. This has been shown to be exceedingly difficult in the case of risk quantification or cost estimation, due to the inherent ethical and computational challenges of evaluating human lives and environmental damages [27, 28], and the discrepancy between the meaning of different interpretations of core terminology, such as *risk*. Even for humans, risk aversion in itself is a highly subjective concept, and it is difficult to conclude upon a universal perception of the term. For example, one may relate (minimum) distances to the size of the vessel, such that larger vessels require larger minimum separation and safe distances. Another suggestion can be to utilize common law rulings to inform what constitutes a safe distance. Until a global consensus is reached and explicit definitions are obtained for these notions, however, the evaluation of risk "costs" is still somewhat abstract, and is consequently treated as such for the time being.

Thus, verification of the resulting risk cost function is approximated through visual presentation and human interpretation of the simulation results presented in this paper. More testing and research is recommended to achieve higher technology readiness levels.

7.4 Safety inequalities and hard constraints

Even though, for example, the safety inequality of **SC1** may be implemented as an explicit (hard) constraint in the constructed control problem, it is argued that hard constraints simultaneously reduce the feasibility and may raise the computational complexity of the NLP to be solved [29]. Thus, the decision was made to relax this constraint and allow it to be violated if the situation calls for it. Note that this subsequently allows for handling of hazardous situations in more complex implementations, i.e., scenarios in which the combined decision-making algorithm deems the cost of purposely grounding the ship favorable to other alternatives (e.g., to avoid a complete loss of the vessel, reduce the probability of an oil spill and/or potential loss of human lives).

Consequently, all explicit inequality constraints were formulated as weighted risk cost, in accordance with Steps 5 to 10 of the proposed method. However, inspection of the other risk cost terms reveals that the act of weighting distances close to grounding obstacles with high values of risk is already achieved by the second (and to some degree the third) term of the constructed risk cost function (12). As such, one may if desired merge the terms if the corresponding cost coefficients are appropriately adjusted such that the cost term strongly and sufficiently discourages violation of the minimum separation distance to grounding obstacles.

This is left to be further explored in future works.

7.5 The structure of the risk cost terms

Making the cost terms monotonic and convex greatly simplifies the complexity of the NLP, leading to faster solving times as well as more predictability with respect to solution quality or expected trajectories (performance). Using various polynomials in place of exponentials was in this context considered, for different levels or ranges of assigned risk priority numbers. However, compared to e.g. x^{-2} or x^{-4}, the exponential function does not contain singularities for inverse proportional relationships such as e^{-x}, making it a more suitable candidate for continuous risk scaling when approaching the safety boundary. Additionally, it is argued that due to limited function domains (e.g., 0–10 km horizon ranges around a ship in a specific environment), there is little to no practical difference between terms of e.g., the form x^{-2} compared to ae^{-bx} for a given domain range and appropriate tuning of the a and b parameters. It is thus assumed that any approximately equivalent behavior may be obtained through the exponential terms alone.

In summary, the first risk term related to **SV1a** is introduced to enforce minimal distances to grounding obstacles by a natural inverse exponential relationship. For **SV5a**, it is assumed that the objective of not grounding is unchanged during loss scenarios with reduced propulsion capabilities, regardless of environmental disturbance forces potentially being dominant. However, **SV5a** serves to encourage even more conservative avoidance distances to grounding obstacles if such scenarios occurs. The magnitude of this effect may be tweaked by adjusting μ_2 and ζ_2 relative to μ_1 and ζ_1. This is due to the fact that by definition, the safety buffer variable d_{safe} of **SI5a** is merely a shifted or more conservative formulation of d_{sep} from **SI1a**. Lastly, the third term of **SV5b** is modified to increase only with positive scalar products between the wind disturbance vector and the vector between the ship and any grounding obstacle σ_j, scaled by the disturbance velocity v_d. This effectively adds an additional safety margin toward down-stream obstacles, which may improve the initial system state if loss scenarios such as machinery faults occurs.

7.6 Simulation performance and parameter tuning

In general, the performance of the MPC is both expected and verified as appropriate. However, the simulation results

are heavily dependent on the specific tuning of parameters as applied in the example demonstrations. Recent research indicate that there exists methods for data-driven or automatic tuning of simple problems [30, 31], in which the latter software is available as an open-source Python package. This is nevertheless a limitation of this study, which means that further evaluation and verification of the simulation performance is necessary.

Similarly, the connection between the assigned RPN and the resulting coefficient values of the exponential risk cost terms presented in the methodology is somewhat abstract. The RPN only provide initial values for parameter tuning, and as such, the cost coefficients must be fine-tuned to each application on a case-to-case basis. Thus, detailed parameter tuning was mostly left out of the scope of the main contribution of this paper.

8 Future work and extensions

8.1 Forward velocity and risks ahead of the ship

In autonomous navigation for surface vessels, the forward velocity and the uncertainties and related risks in the front of the ship are considered as significant contributions to the risk of a given situation, potentially dominating other risk factors, such as the uncertainty and grounding risk related to lateral on-land wind disturbances. It is recommended that this is addressed in future implementations.

8.2 Machinery system additions

Additional risk cost terms for wear and tear or component failure in the machinery system related to high-intensity operation periods or over-use may be included in future versions of the implementation, including certain thresholds or dynamically weighted costs for machinery utilization. With varying AMM modes, the machinery system may experience changes in its inherent uncertainty and probability estimations for, e.g., a blackout scenario based on various available system modes or power configurations. The margins could be smaller with safer machinery modes, and the ship may in such situations consequently sail closer to land. Fuel consumption modeled in the machinery model may similarly also yield a more thorough understanding of the actual costs related to various control actions.

8.3 NLP solver considerations

Due to the complexity of the environment (i.e., sea depth polygon obstacles provided by an ENC module) in the constructed NLP, feasible solutions that successfully carries out the given mission are not guaranteed. Moreover, if the solver is not able to converge to a solution within the given maximum time limit, the returned solution may be dangerous or even physically impossible with respect to the defined ship dynamics. In this case, the MPC could fail to produce a suitable trajectory, and this is regarded as a drawback to this method. A potential remedy to this challenge may be to, e.g., employ the use of an additional backup controller and a performance monitor to assume emergency control or make the human intervene in the ROC if failures or problems are detected in the MPC, as suggested by recent research [32].

Moreover, the complexity or resolution of the mapped grounding obstacles constructed during initialization of the ENC, as well as the discretization step or resolution of the ship trajectory (i.e., sampling time) are of significant importance with respect to the performance of the NLP solver. Lower spatial and temporal resolutions reduce the time complexity of the ENC minimum distance computations, but also decrease the accuracy and confidence of the computed optimal solution. An appropriate balance between these essential factors is in general difficult to determine, and must in addition to the cost function parameter tuning be established and verified on a case-by-case basis.

As a result of the non-convexity of the ENC grounding obstacles and their respective risk cost, no global minimum solution is guaranteed. Hence, providing a suitable initial guess to the NLP solver during setup is critical in order to both achieve adequate solver performance and to ensure feasibility of the optimal NLP solutions. It is thus necessary that a conservative initial guess is properly constructed such that the solution converges to an appropriate local minimum, with respect to the expected trajectory of the navigational mission given to the ship. Due to warm start, subsequent initial guesses are provided to the solver as the forward shifted solution of the previous solve, and are consequently also largely dependent on the solution guess of the very first solve. Furthermore, this initial guess should not diverge from the optimal solution to such a degree that the solver is not able to calculate and return the solution within the required time interval. It is proposed that evolutionary or genetic algorithms such as particle swarm optimization may serve as a possible alternative approach for this problem, in order to obtain global convergence less dependent on the initial conditions of the NLP.

8.4 Safety framework and risk model utilization

An additional consideration may be to transform or model tuning variables or cost coefficients into a safety framework, or to employ the use of an appropriate risk model. Recent research has shown that scenario-based MPC may utilize a probabilistic uncertainty model to achieve safe path traversal for e.g., inspection drones [33]. This may prove useful in order further structure the considered problem, to speed up

the tuning process, and to enable use of models for resource limited embedded and real-time computing.

8.5 Other improvements

No sensor uncertainties were considered in this study, and may be implemented in future works. Additionally, parallel scenario simulations may be utilized during runtime to predict more complex risk pictures for any given time instant, beyond the current system state and environment conditions considered in this work. Collision avoidance and COLREGs handling are considered natural features of autonomous navigation systems, and should be included in future works. It is lastly recommended that more scenarios are investigated for analysis and simulation purposes in order to further increase the robustness and reliability of the MPC scheme.

9 Conclusion

A systematic and novel method has been proposed that enables the use the results of risk analysis to formulate an optimizable supervisory risk control problem through MPC, taking safety constraints and risk factors systematically into account. The risk analysis in the paper was performed using STPA with a focus on anti-grounding for an autonomous ship. A method providing appropriate system state variables and equations and a risk-based cost function for an optimal control problem, based on the STPA results has been proposed. The optimal control problem was subsequently transformed into a nonlinear program and solved using an MPC scheme with a receding horizon approach. Several demonstrated control scenarios for an autonomous ship, simulated by an MPC scheme, show that the proposed method for construction of quantitative and optimizable risk-based costs based on safety constraints from STPA produces adequate and safe control trajectories. Additionally, the analysis has identified some vulnerabilities that should be addressed in future works. Ultimately, the paper shows that constructing the MPC objective function based on the results from STPA produces ANS behavior appropriate for safe navigation of ships, thus supporting the hypothesis that increased levels of safety may be achieved by the MPC-based ANS through systematic analysis of unsafe control actions and hazards when designing the MPC cost function. This approach is consequently considered a reasonable bridge between the realms of qualitative risk analysis and numerical optimal control.

10 Appendix: Ship model and dynamics

To model the risk and enable optimization and control of related physical processes, the mathematical and physical relationships between the autonomous ship and its environment are formulated. This section defines the ship model used in this work, adapted from the ship model and the terminology as presented in [34].

The horizontal plane North-East (NE) and BODY coordinate frames are defined as given in Fig. 12. The NE coordinate system assumes a locally flat ocean surface plane and is oriented with its X- and Y-axes toward the true North and East, respectively. The BODY reference frame is positioned with its origin located in the centroid of the ship.

Given the previous reference frame definitions, the model variables are defined according to Fig. 12:

$$\text{NE position} \quad \boldsymbol{p}^n_{b/n} = \begin{bmatrix} x \\ y \end{bmatrix} \in \mathbb{R}^2$$

$$\text{Body-fixed linear velocity} \quad \boldsymbol{v}^b_{b/n} = \begin{bmatrix} u \\ v \end{bmatrix} \in \mathbb{R}^2$$

$$\text{Body-fixed propulsion forces} \quad \boldsymbol{f}^b_b = \begin{bmatrix} X \\ Y \end{bmatrix} \in \mathbb{R}^2$$

$$\text{Attitude (yaw angle)} \quad \boldsymbol{\Theta}_{nb} = \begin{bmatrix} \psi \end{bmatrix} \in \mathbb{R}$$

$$\text{Body-fixed angular velocity} \quad \boldsymbol{\omega}^b_{b/n} = \begin{bmatrix} r \end{bmatrix} \in \mathbb{R}$$

$$\text{Body-fixed rotational moment} \quad \boldsymbol{m}^b_b = \begin{bmatrix} N \end{bmatrix} \in \mathbb{R}$$

where $\{n\}$ is the NE reference frame and $\{b\}$ is the BODY reference frame. The ship states, forces and moments are defined by the variables

$$\boldsymbol{\eta} = \begin{bmatrix} \boldsymbol{p}^n_{b/n} \\ \boldsymbol{\Theta}_{nb} \end{bmatrix}, \quad \boldsymbol{\nu} = \begin{bmatrix} \boldsymbol{v}^n_{b/n} \\ \boldsymbol{\omega}^b_{b/n} \end{bmatrix}, \quad \boldsymbol{\tau} = \begin{bmatrix} \boldsymbol{f}^b_b \\ \boldsymbol{m}^b_b \end{bmatrix} \tag{17}$$

where $\boldsymbol{\eta}$, $\boldsymbol{\nu}$ and $\boldsymbol{\tau}$ denotes the position, velocity and forces or moments vectors in the horizontal plane, i.e. surge, sway and

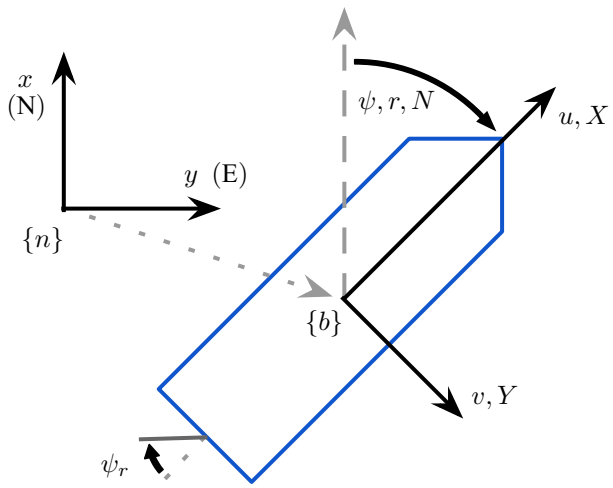

Fig. 12 The model variables and coordinate frames of the autonomous ship used in this work

yaw, respectively. Moreover, the principal rotation matrix in the XY-plane is defined as

$$\boldsymbol{R}_b^n(\boldsymbol{\Theta}_{nb}) = \begin{bmatrix} \cos(\psi) & -\sin(\psi) \\ \sin(\psi) & \cos(\psi) \end{bmatrix} \in \mathbb{R}^{2\times 2} \tag{18}$$

Due to the roll and pitch angles being neglected, the body-fixed velocity vectors can be expressed in $\{n\}$ as

$$\dot{\boldsymbol{p}}_{b/n}^n = \boldsymbol{R}_b^n(\boldsymbol{\Theta}_{nb})\boldsymbol{v}_{b/n}^b, \quad \dot{\boldsymbol{\Theta}}_{nb} = \boldsymbol{\omega}_{b/n}^b \tag{19}$$

The Jacobian $\boldsymbol{J}_{\Theta}(\boldsymbol{\eta})$ is further given by

$$\boldsymbol{J}_{\Theta}(\boldsymbol{\eta}) = \begin{bmatrix} \boldsymbol{R}_b^n(\boldsymbol{\Theta}_{nb}) & \boldsymbol{0} \\ \boldsymbol{0} & 1 \end{bmatrix} \tag{20}$$

and the resulting kinematic equations are formulated as

$$\dot{\boldsymbol{\eta}} = \boldsymbol{J}_{\Theta}(\boldsymbol{\eta})\boldsymbol{\nu} \tag{21}$$

The reduced three-dimensional ship kinetics equations in the horizontal XY-plane (with no Coriolis, wave, ballast, buoyancy or gravitational forces included) are given by

$$(\boldsymbol{M}_{RB} + \boldsymbol{M}_A)\dot{\boldsymbol{\nu}} + \boldsymbol{D}\boldsymbol{\nu} = \boldsymbol{\tau} + \boldsymbol{\tau}_{\text{wind}} + \boldsymbol{\tau}_{\text{currents}} \tag{22}$$

where $\boldsymbol{M}_{RB}$ is the rigid body mass matrix, $\boldsymbol{M}_A$ is the hydrodynamic added mass matrix, $\boldsymbol{\tau}_{\text{wind}}$ and $\boldsymbol{\tau}_{\text{currents}}$ are the wind and currents forces, and $\boldsymbol{D}$ is a constant damping matrix. In the example simulations presented in this work, $\boldsymbol{\tau}_{\text{currents}} = 0$ for simplicity, and the wind forces $\boldsymbol{\tau}_{\text{wind}}$ are defined according to [34] as

$$\boldsymbol{\tau}_{\text{wind}} = \begin{bmatrix} X_{\text{wind}} \\ Y_{\text{wind}} \\ N_{\text{wind}} \end{bmatrix} = \begin{bmatrix} -c_x \cos(\Psi_w) A_{Fw} \\ c_y \sin(\Psi_w) A_{Lw} \\ c_n \sin(2\Psi_w) A_{Lw} L_{oa} \end{bmatrix} \frac{1}{2}\rho_a V_w^2 \tag{23}$$

where V_w is the relative wind velocity with respect to the ship's velocity, $\Psi_w = \psi - \psi_w - \pi$, ψ_w is the clockwise wind angle relative to the North axis, and the wind coefficients c_* are generated by polynomial approximations of a wind coefficient table for a given ship.

Lastly, the propulsion and steering forces vector is given as

$$\boldsymbol{\tau}(\psi_r) = \begin{bmatrix} u_1 \\ 0 \\ -f_{\text{rudder}}(.)u^2 \sin(\psi_r) \end{bmatrix} \tag{24}$$

given the definitions from Fig. 12, where $f_{\text{rudder}}(.)$ is a rudder coefficient function, u is the forward surge velocity, and ψ_r is the rudder angle.

See [34] for generalizations to other propulsion and steering configurations.

Acknowledgements This work was carried out with the helpful guidance of associates within the Online risk management and risk control for autonomous ships (ORCAS) project funded by the Research Council of Norway (Grant No. 280655), Kongsberg Maritime and DNV GL, and Centre for Autonomous Marine Operations and Systems (AMOS) at NTNU funded by the Research Council of Norway (Grant No. 223254). Moreover, many thanks to all of the participants of the internal STPA workshop, which provided the basis for the risk analysis presented in this work. Finally, the authors would like to express their gratitude to the anonymous reviewers for their invaluable comments on this paper.

Funding Open access funding provided by NTNU Norwegian University of Science and Technology (incl St. Olavs Hospital - Trondheim University Hospital).

Data availability The data that support the findings of this study are available from the corresponding author, S. B., upon reasonable request.

References

1. Utne IB, Schjølberg I, Roe E (2019) High reliability management and control operator risks in autonomous marine systems and operations. Ocean Eng 171:399–416
2. Shappell SA, Wiegmann DA (2000) The Human Factors Analysis and Classification System-HFACS, United States. Office of Aviation Medicine, Tech Rep.
3. Ramos MA, Utne IB, Mosleh A (2019) Collision avoidance on maritime autonomous surface ships: operators' tasks and human failure events. Saf Sci 116:33–44
4. Layman L, Basili VR, Zelkowitz MV (2014) A methodology for exposing risk in achieving emergent system properties. ACM Trans Softw Eng Methodol 23(3):221–2228. https://doi.org/10.1145/2560048
5. Øien K, Utne I, Herrera I (2011) Building safety indicators: part 1—theoretical foundation. Saf Sci 49(2):148–161 (Online). Available: http://www.sciencedirect.com/science/article/pii/S0925753510001335
6. Øien K, Utne I, Tinmannsvik R, Massaiu S (2011) Building safety indicators: part 2—application, practices and results. Saf Sci 49(2):162–171 (Online). Available: http://www.sciencedirect.com/science/article/pii/S0925753510001360
7. Bø TI, Johansen TA (2016) Dynamic safety constraints by scenario-based economic model predictive control of marine electric power plants. IEEE Trans Transp Electrif 3(1):13–21
8. Veksler A, Johansen TA, Borrelli F, Realfsen B (2016) Dynamic positioning with model predictive control. IEEE Trans Control Syst Technol 24(4):1340–1353

9. Zhou H, Guvenc L, Liu Z (2017) Design and evaluation of path following controller based on MPC for autonomous vehicle. In: Chinese control conference, CCC, pp 9934–9939
10. Eriksen BOH, Breivik M (2017) MPC-Based mid-level collision avoidance for ASVs using nonlinear programming. In: 1st annual IEEE conference on control technology and applications, CCTA 2017, vol 2017-Janua, pp 766–772
11. Kufoalor DKM, Johansen TA, Brekke EF, Hepsø A, Trnka K (2019) Autonomous maritime collision avoidance: field verification of autonomous surface vehicle behavior in challenging scenarios. J Field Robot. https://doi.org/10.1002/rob.21919
12. Utne IB, Rokseth B, Sørensen AJ, Vinnem JE (2020) Towards supervisory risk control of autonomous ships. Reliab Eng Syst Saf 196:106757
13. Blindheim S, Gros S, Johansen TA (2020) Risk-based model predictive control for autonomous ship emergency management. IFAC Pap OnLine 53(2):14524–14531
14. Rausand M (2013) Risk assessment: theory, methods, and applications, vol 115. Wiley, London
15. Utne IB, Sørensen AJ, Schjølberg I (2017) Risk management of autonomous marine systems and operations. In: International conference on offshore mechanics and arctic engineering, vol 57663. American Society of Mechanical Engineers, p V03BT02A020
16. Ludvigsen M, Sørensen AJ (2016) Towards integrated autonomous underwater operations for ocean mapping and monitoring. Annu Rev Control 42:145–157
17. Leveson N (2011) Engineering a safer world: systems thinking applied to safety. MIT Press, London
18. Rokseth B, Utne IB, Vinnem JE (2017) A systems approach to risk analysis of maritime operations. Proc Inst Mech Eng Part O J Risk Reliab 231(1):53–68
19. Rokseth B, Haugen OI, Utne IB (2019) Safety verification for autonomous ships. In: MATEC web of conferences, vol 273. EDP Sciences, p 02002
20. Leveson NG, Thomas JP (2018) Stpa handbook. Cambridge University Press, Cambridge
21. Stopford M (2009) Maritime economics, 3rd edn. Routledge, London
22. Blindheim S, Johansen TA (2021) Electronic navigational charts for visualization, simulation, and autonomous ship control. IEEE Access 10:3716–3737
23. Kim H, Lundteigen MA, Hafver A, Pedersen FB (2021) Utilization of risk priority number to systems-theoretic process analysis: a practical solution to manage a large number of unsafe control actions and loss scenarios. Proc Inst Mech Eng Part O J Risk Reliab 235(1):92–107
24. Kiran D (2017) Chapter 26—failure modes and effects analysis. In: Kiran D (ed) Total quality management. Butterworth-Heinemann, pp 373–389 (Online). Available: https://www.sciencedirect.com/science/article/pii/B978012811035500026X
25. Wright S, Nocedal J (1999) Numerical optimization. Springer Sci 35(67–68):7
26. Morrison DD, Riley JD, Zancanaro JF (1962) Multiple shooting method for two-point boundary value problems. Commun ACM 5(12):613–614
27. Mishan EJ (1971) Evaluation of life and limb: a theoretical approach. J Polit Econ 79(4):687–705. https://doi.org/10.1086/259784
28. Ackerman F, Heinzerling L (2002) Pricing the priceless: cost-benefit analysis of environmental protection. University of Pennsylvania Law Rev 150(5):1553–1584 (Online). Available: http://www.jstor.org/stable/3312947
29. Richards A (2015) Fast model predictive control with soft constraints. Eur J Control 25:51–59
30. Alhajeri M, Soroush M (2020) Tuning guidelines for model-predictive control. Ind Eng Chem Res 59(10):4177–4191
31. Edwards W, Tang G, Mamakoukas G, Murphey T, Hauser K (2021) Automatic tuning for data-driven model predictive control. In: 2021 IEEE international conference on robotics and automation (ICRA). IEEE, pp 7379–7385
32. Johansen TA (2017) Toward dependable embedded model predictive control. IEEE Syst J 11(2):1208–1219
33. Rothmund SV, Johansen TA (2019) Risk-based obstacle avoidance in unknown environments using scenario-based predictive control for an inspection drone equipped with range finding sensors. In: 2019 international conference on unmanned aircraft systems (ICUAS). IEEE, pp 221–230
34. Fossen TI (2011) Handbook of marine craft hydrodynamics and motion control. Wiley, London