Preprints of the 21st IFAC World Congress (Virtual)
Berlin, Germany, July 12-17, 2020

Risk-Based Model Predictive Control for Autonomous Ship Emergency Management

Simon Blindheim * **Sebastien Gros** ** **Tor Arne Johansen** *

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

Abstract: Control for semi- and fully-autonomous ships is a broad and complex field. Making autonomous high-level decisions in place of the captain is considered difficult, partly due to the risks and uncertainties involved. Though human operators located in onshore control centers are still needed for safety and regulatory reasons, there is a growing demand for complex decisions to be made by the onboard control system itself, both during normal operations and extraordinary circumstances. Model predictive control (MPC) is a promising approach to tackle this problem. In this paper, a dynamic risk-based decision-making algorithm is constructed through the use of heuristic objectives, capable of planning suitable vessel trajectories in emergency situations. Nonlinear programming using the direct multiple-shooting method implemented with the CasADi framework is considered, and the resulting control performance of several emergency scenarios is analyzed using simulation. The developed algorithm proved capable of both generating suitable trajectories below a certain risk threshold, and to engage the safety systems appropriately. It is concluded that MPC with independent risk cost terms is a promising method for autonomous ship trajectory planning and emergency management.

Keywords: Model Predictive Control, Risk Control, Autonomous Control, Decision-Making, Emergency Management, Trajectory Planning, Online Optimization, Maritime Systems

1. INTRODUCTION

This paper focuses on determining sequences of control actions to be taken in maritime emergency situations, in which it is not deemed appropriate to – or the ship is not able to – operate normally. The motivation behind this work is the ever-increasing desire to further reduce both operational costs and risks during shipping operations, particularly by moving personnel normally on board the vessel to onshore control centers. To achieve this, ships need increased autonomy and onboard decision-making capabilities. Model predictive control (MPC) has shown great results for autonomous vehicle steering (Keviczky et al., 2006), ship heading control (Li and Sun, 2012), path following (Zhou et al., 2017) and collision avoidance (Eriksen and Breivik, 2017), (Kufoalor et al., 2019). However, these systems usually have strictly defined operational constraints or limited available decision spaces in which they are explicitly allowed to make autonomous decisions. Conditions such as these are normally the default operational stages, like the transit phase of a fjord-crossing autonomous ferry. In order to reach higher levels of autonomy, a more high-level supervisory system for risk or threat assessment and decision-making (Anderson et al., 2011), (Samuelson and Yang, 2018), (Chen et al., 2018) for maritime operations is needed. Thus, the main purpose of this paper is to investigate the use of MPC for handling emergency situations that are normally taken care of by human operators, through the use of some risk model and optimization-based decision-making. The approach is summarized as follows: The ship control is performed using a receding horizon approach, based on a dynamic ship model, a cost function and operational constraints. Each term in the constructed cost function targets different aspects of trajectory planning during normal operations and various emergency scenarios. Specifically, the developed algorithm handles a selection of abnormal or hazardous operational situations in which some degree of uncertainty is involved. As such, the novel contribution of this paper is to include a separate *risk* term as an additional optimization cost, which makes it possible to address the uncertainty inherent in emergency scenarios directly. This term combined with other costs for resource management and mission objectives may collectively form a total emergency management algorithm, capable of handling all of the presented scenarios simultaneously. The resulting risk-based decision-making method may in turn serve as a foundation for a decision support system for human operators and as an autonomous navigation system for fully autonomous vessels.

2. PROBLEM DESCRIPTION

2.1 Scope and simplifications

The main objective of this work is to control the trajectory of a ship along a preplanned path in a challenging maritime environment, demonstrated by simulating a crossing

through a strait with grounding obstacles on both sides. A simplified dynamic environment is used in this work, in which only variables related to the horizontal movement of the ship position are considered. The ship model is equipped with two freely rotating azimuth thrusters (one at the bow and one at the stern) with given maximum power specifications, and wind and currents velocities are assumed to be constant. No collision avoidance or sensor data quality handling is considered in this work, as these concepts are assumed to be added as natural extensions in a more exhaustive system (Johansen et al., 2016). Docking is also disregarded here, as it may be viewed as a separate control mode. Lastly, the approach presented in this paper assumes that the ship and/or the operator is able to stop or react quickly when the risk is too large. However, it is considered a trivial task to appropriately increase the related risk coefficients to account for the stopping dynamics of the anchor drop or other significant delays as a consequence of higher velocities. These simplifications and approximations are used to develop a simple model serving as a proof of concept.

2.2 Failure modes and emergency scenarios

A collection of scenarios are presented in Table 1 to showcase the proposed method in this work:

1. *Impaired thrusters*
In this failure mode, the propulsion system has reduced maneuvering capabilities. No wind disturbances are assumed. **a)** Both thrusters lose the ability to rotate for a period of time, leaving the ship with constant thruster azimuth angles. Steering along the path is achieved by changing thrust magnitude only, until azimuth rotation capabilities are restored. **b)** The bow thruster goes offline. Thus the MPC scheme must use reduced degrees of freedom, i.e. the stern thruster only, to complete its mission.
2. *Total blackout*
The ship experiences a complete loss of propulsion due to a *temporary* power blackout, until the crew is able to restart the engines. Moderate wind disturbances lead to drifting. If and when to drop the anchor is continuously assessed by the algorithm. **a)** If the ship recovers its propulsion capabilities before the grounding hazard is too large, an alternative trajectory is calculated after drifting away from the original path. **b)** An anchor drop is triggered if the maximum grounding risk threshold becomes violated.
3. *Strong winds*
The increased grounding risks due to exceedingly strong winds are assessed in order to perform sufficiently safe control actions. **a)** A reference scenario demonstrates how the added risk term contributes to adjustments in the ship trajectory close to grounding obstacles. **b)** Crossing the strait is deemed too dangerous due to strong winds. As a result, the ship holds its position and waits for improved weather conditions for some time. **c)** The ship avoids the narrow strait in its entirety and opts to navigate around the nearby smaller isles, as a result of an alternative risk cost tuning approach.

Table 1. Demonstration scenarios

3. MATHEMATICAL MODELING

3.1 Variables and reference frame definitions

First, the locally flat North-East (NE) coordinate frame $\{n\}$ and the body coordinate frame $\{b\}$ are defined as presented in Figure 1. The variables are defined as follows: x and y denote the position of the ship along the North and East axes, u and v are the surge and sway velocities of the ship, X and Y are the surge and sway forces of the ship, ψ, r and N are the yaw angle, velocity and moment of the ship, and $\boldsymbol{a}_t = [a_1 \ a_2]^\top$ and $\boldsymbol{f}_t = [f_1 \ f_2]^\top$ are the azimuth angles and propulsion forces of the ship's stern and bow thrusters, respectively.

3.2 Ship model and dynamics

The model variables are given in Table 2. From (2.1) and (2.2) in Fossen (2011), the reduced three-dimensional ship kinematic and kinetics equations in the horizontal NE-plane (disregarding Coriolis, wave, ballast, buoyancy or gravitational forces) are given as

$$\dot{\boldsymbol{\eta}} = \boldsymbol{J}_{\Theta}(\boldsymbol{\eta})\boldsymbol{\nu} \tag{1}$$

$$\dot{\boldsymbol{\nu}} = \boldsymbol{M}^{-1}(\boldsymbol{\tau} + \boldsymbol{d} - \boldsymbol{D}\boldsymbol{\nu}) \tag{2}$$

where $\boldsymbol{d} = \boldsymbol{\tau}_{\text{wind}} + \boldsymbol{\tau}_{\text{current}}$ is the system disturbance vector. The ocean currents forces $\boldsymbol{\tau}_{\text{current}} \triangleq 0$ in the example simulations presented in this work, for simplicity. The wind forces are defined as

$$\boldsymbol{\tau}_{\text{wind}} = \begin{bmatrix} -c_x \cos(\gamma_w) A_{Fw} \\ c_y \sin(\gamma_w) A_{Lw} \\ c_n \sin(2\gamma_w) A_{Lw} L_{oa} \end{bmatrix} \frac{1}{2}\rho_a V_w^2 \tag{3}$$

from Fossen (2011), where V_w is the wind velocity relative to the ship's velocity, $\gamma_w = \psi - \psi_w - \pi$, and ψ_w is the clockwise wind angle relative to the North axis. The wind coefficients c_x, c_y and c_n are in this work set to 0.7, 0.8 and 0.1, respectively. See Section 5.3 for all remaining model parameter definitions and their given values, and see Fossen (2011) for generalizations to other propulsion and steering configurations.

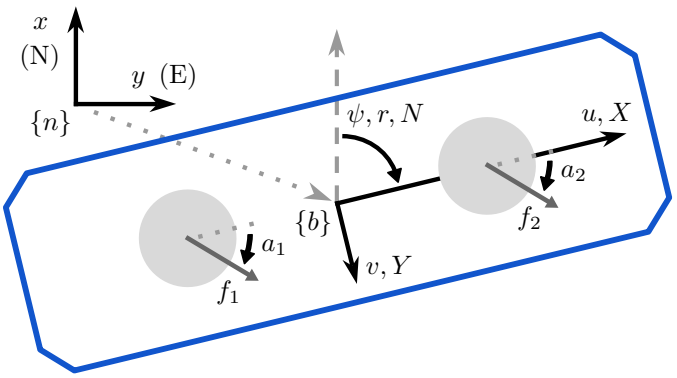

Fig. 1. The model variables and coordinate frames used in this work.

Entity	Symbol	Elements
North-East ship position	$\boldsymbol{p}^n_{b/n}$	$\begin{bmatrix} x \\ y \end{bmatrix}$
North-East ship attitude	$\boldsymbol{\Theta}_{nb}$	$\begin{bmatrix} \psi \end{bmatrix}$
Ship position and orientation	$\boldsymbol{\eta}$	$\begin{bmatrix} \boldsymbol{p}^n_{b/n} \\ \boldsymbol{\Theta}_{nb} \end{bmatrix}$
Body-fixed linear velocity	$\boldsymbol{v}^b_{b/n}$	$\begin{bmatrix} u \\ v \end{bmatrix}$
Body-fixed angular velocity	$\boldsymbol{\omega}^b_{b/n}$	$\begin{bmatrix} r \end{bmatrix}$
Linear and angular ship velocities	$\boldsymbol{\nu}$	$\begin{bmatrix} \boldsymbol{v}^b_{b/n} \\ \boldsymbol{\omega}^b_{b/n} \end{bmatrix}$
System state vector	$\boldsymbol{x}$	$\begin{bmatrix} \boldsymbol{\eta} \\ \boldsymbol{\nu} \end{bmatrix}$
Principal rotation matrix	$\boldsymbol{R}^n_b(\boldsymbol{\Theta}_{nb})$	$\begin{bmatrix} \cos(\psi) & -\sin(\psi) \\ \sin(\psi) & \cos(\psi) \end{bmatrix}$
Ship pose Jacobian	$\boldsymbol{J}_\Theta(\boldsymbol{\eta})$	$\begin{bmatrix} \boldsymbol{R}^n_b(\boldsymbol{\Theta}_{nb}) & \boldsymbol{0} \\ \boldsymbol{0} & 1 \end{bmatrix}$
North-East linear velocity	$\dot{\boldsymbol{p}}^n_{b/n}$	$\boldsymbol{R}^n_b(\boldsymbol{\Theta}_{nb})\boldsymbol{v}^b_{b/n}$
North-East angular velocity	$\dot{\boldsymbol{\Theta}}_{nb}$	$\boldsymbol{\omega}^b_{b/n}$
Thrusters transformation matrix $s_* \coloneqq \sin(a_*) \quad c_* \coloneqq \cos(a_*)$	$\boldsymbol{T}(\boldsymbol{a}_t)$	$\begin{bmatrix} c_1 & c_2 \\ s_1 & s_2 \\ -l_x s_1 & l_x s_2 \end{bmatrix}$
Body-fixed propulsion forces	$\boldsymbol{f}^b_b$	$\begin{bmatrix} X \\ Y \end{bmatrix}$
Body-fixed moment (torque)	$\boldsymbol{m}^b_b$	$\begin{bmatrix} N \end{bmatrix}$
Ship forces and moments	$\boldsymbol{\tau}$	$\boldsymbol{T}(\boldsymbol{a}_t)\boldsymbol{f}_t = \begin{bmatrix} \boldsymbol{f}^b_b \\ \boldsymbol{m}^b_b \end{bmatrix}$
Control input vector	$\boldsymbol{u}$	$\begin{bmatrix} \boldsymbol{f}_t \\ \dot{\boldsymbol{a}}_t \end{bmatrix}$
Constant damping matrix	$\boldsymbol{D}$	$\mathrm{diag}(X_u, Y_v, N_r)$
Hydrodynamic added mass	$\boldsymbol{M}_A$	$\mathrm{diag}(X_{\dot{u}}, Y_{\dot{v}}, N_{\dot{r}})$
Rigid-body ship mass	$\boldsymbol{M}_{RB}$	$\mathrm{diag}(m, m, I_z)$
Total model mass	$\boldsymbol{M}$	$\boldsymbol{M}_{RB} + \boldsymbol{M}_A$

Table 2. Model terminology and definitions

3.3 Optimal control problem formulation

An optimal control problem (OCP) is defined as follows:

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

where $\tilde{\phi}$ is a scalar stage cost function, $\boldsymbol{\theta}$ is a parameter vector, $\boldsymbol{x}_0$ is the initial state, T is the prediction horizon, and $\dot{\boldsymbol{x}}$ is given by the system dynamics (1) and (2). The constraints $\boldsymbol{h}(\boldsymbol{x}(t), \boldsymbol{u}(t))$ are given as:

$$\begin{aligned} -f_{\max} &\leq u_1 \leq f_{\max} \\ -f_{\max} &\leq u_2 \leq f_{\max} \\ -\omega_{\max} &\leq u_3 \leq \omega_{\max} \\ -\omega_{\max} &\leq u_4 \leq \omega_{\max} \end{aligned} \tag{5}$$

where $\boldsymbol{u} = [u_1\ u_2\ u_3\ u_4]^\top$ from Table 2, and $f_{\max}$ and $\omega_{\max}$ are the maximum propulsion force and rotational turning rate of the azimuth thrusters, respectively. The solution to problem (4) will be deployed in a receding horizon fashion, yielding an MPC scheme.

3.4 Nonlinear programming

Next, the model is discretized in order to solve the problem numerically. The continuous time variable t is divided into a time grid of N intervals, defined by discrete time instants $t_k \in \{t_0, t_1, ..., t_N\}$. 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 (NLP) by defining a vector of decision variables

$$\boldsymbol{w} = \begin{bmatrix} \boldsymbol{x}_0^\top\ \boldsymbol{q}_0^\top\ \boldsymbol{u}_0^\top\ \ldots\ \boldsymbol{x}_{N-1}^\top\ \boldsymbol{q}_{N-1}^\top\ \boldsymbol{u}_{N-1}^\top\ \boldsymbol{x}_N^\top\ \boldsymbol{q}_N^\top \end{bmatrix}^\top \tag{6}$$

where $\boldsymbol{q}_k$ is a vector of additional decision variables related to mission objectives to be defined in Section 4.1. 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 only parameters considered in this work are

$$\boldsymbol{\theta}_k = \begin{bmatrix} s_{\text{ref}} \\ \alpha_{\text{step}} \\ \boldsymbol{\sigma} \end{bmatrix} \in \mathbb{R}^{2+3J} \tag{7}$$

for each t_k where s_{ref} is a constant reference transit speed and α_{step} is a path progression parameter (see Section 4) across all N control intervals. The grounding obstacles are modeled as a union of J circles. Thus, $\boldsymbol{\sigma}$ is the collection of all grounding hazard vectors of the form

$$\boldsymbol{\sigma}_j = \begin{bmatrix} \boldsymbol{c}_j \\ r_j \end{bmatrix}, \qquad \boldsymbol{c}_j = \begin{bmatrix} x_j \\ y_j \end{bmatrix}, \qquad j = 1, ..., J \tag{8}$$

where $\boldsymbol{c}_j$ and r_j are the center point and radius of each obstacle, respectively. The resulting NLP is defined as

$$\begin{aligned} C(\boldsymbol{\theta}, \boldsymbol{x}_0) = \min_{\boldsymbol{w}} \quad & \phi(\boldsymbol{w}, \boldsymbol{\theta}) \\ \text{s.t.} \quad & \boldsymbol{g}(\boldsymbol{w}) = \boldsymbol{0} \\ & \boldsymbol{h}(\boldsymbol{w}) \leq \boldsymbol{0} \end{aligned} \tag{9}$$

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 $\boldsymbol{x}_0$. The inequality constraints $\boldsymbol{h}(\boldsymbol{w})$ are given by (5), and the equality constraints $\boldsymbol{g}(\boldsymbol{w})$ hold the system dynamics:

$$\boldsymbol{F}_k(\boldsymbol{x}_k, \boldsymbol{u}_k, \boldsymbol{d}_k) - \boldsymbol{x}_{k+1} = \boldsymbol{0}, \qquad k = 0, 1, ..., N-1 \tag{10}$$

The cost function ϕ is defined and discussed in Section 4.2. Note that the discretization chosen here is based on the *direct multiple-shooting* approach (Morrison et al., 1962). Because of the nonlinear dynamics and since the obstacles yield a non-convex feasible set, the NLP (9) 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. The preplanned path and ship speed reference parameters are used to calculate a reasonable initial guess for the ship trajectory. Rather than using hard constraints in addition to the ship dynamics and the natural input constraints, only costs balancing is utilized to achieve the desired control behavior. This ensures feasibility of the NLP solutions.

4. PLANNING AND DECISION-MAKING

The decision-making of the MPC algorithm consists of two main tasks: the planning of the ship trajectory achieved by propulsion and steering control, and the decision to drop the anchor in emergency situations. If the planning algorithm is unable to produce a trajectory that does not violate the given grounding risk thresholds, emergency procedures are triggered in order to minimize damages and costs – such as dropping the anchor. It may be noted that this can also be achieved by a formulation based on mixed-integer programming, allowing for more complex decisions to be made during emergency situations. The following subsections describe the chosen path-following method, as well as how different mission objectives are evaluated and weighted in order to produce desired ship trajectories.

4.1 The path following method

There are many different methods readily available for following a preplanned path, e.g. by using a predefined stride each time interval along the path, or a line-of-sight method (Fossen et al., 2003). A time-invariant method is chosen in this work to generate consistent and robust solutions at any time interval. First, a preplanned path is chosen by designing a piecewise linear (spline) function given an initial position, discrete intermediate points and a destination. It is assumed that the path is designed such that fuel/resource consumption, time spent and distance traveled is considered (close to) optimal for the given mission. Next, the reference path is parameterized, giving the two-dimensional reference function

$$\boldsymbol{r}(\alpha) = \begin{bmatrix} x(\alpha) \\ y(\alpha) \end{bmatrix} \tag{11}$$

for calculating path points where $\alpha \geq 0$ is an advancement parameter acting as a decision variable along the preplanned path, and $x(\alpha)$ and $y(\alpha)$ are piecewise linear functions. As such, advancing along the path is a simple matter of increasing α. 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

$$u^2 + v^2 \leq s_{\text{ref}}^2 + \beta, \qquad 0 \leq \beta \tag{12}$$

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

$$\boldsymbol{q}_k = \begin{bmatrix} \boldsymbol{a}_t \\ \alpha_k \\ \beta_k \end{bmatrix}, \qquad k = 0, 1, ..., N-1 \tag{13}$$

and the NLP decision variable vector $\boldsymbol{w}$ is well defined.

4.2 Objectives and cost function definitions

In order to complete the NLP, a cost function to be minimized is constructed. In this research, the cost function is heuristically defined with the purpose of producing a safe ship trajectory that fulfills the mission objectives. The primary cost function is denoted as

$$\phi(\boldsymbol{w}, \boldsymbol{\theta}) = \sum_{k=1}^{N} \xi(\boldsymbol{x}_k, \boldsymbol{q}_k, \boldsymbol{q}_{k\text{-}1}) + \epsilon(\boldsymbol{u}_k, \boldsymbol{u}_{k\text{-}1}) + \rho(\boldsymbol{x}_k, \boldsymbol{\theta}_k) \tag{14}$$

where the individual cost terms are defined as follows:

i. The path progression cost function

$$\xi(\boldsymbol{x}_k, \boldsymbol{q}_k, \boldsymbol{q}_{k-1}) = \boldsymbol{\kappa}^\top \begin{bmatrix} ||\boldsymbol{r}(\alpha_k) - \boldsymbol{p}_k||^2 \\ ||\alpha_k - \alpha_{k-1} - \alpha_{\text{step}}||^2 \\ \beta_k \end{bmatrix} \tag{15}$$

where $\boldsymbol{\kappa} > \boldsymbol{0}$. These terms are responsible for driving the ship position $\boldsymbol{p}_k$ (trajectory) along the precomputed feasible path, through the constant path step parameter α_{step} and the reference function $\boldsymbol{r}(\alpha_k)$. The β_k term penalizes violations of the transit speed reference as detailed in Section 4.1. It is recommended that α_{step} is chosen such that $s_{\text{ref}} \approx \alpha_{\text{step}}/t_\Delta$, where t_Δ is the sampling period of the NLP.

ii. Next, the control input cost function is defined as

$$\epsilon(\boldsymbol{u}_k, \boldsymbol{u}_{k-1}) = \boldsymbol{u}_k^\top \boldsymbol{\Lambda} \boldsymbol{u}_k + (\boldsymbol{u}_k - \boldsymbol{u}_{k-1})^\top \boldsymbol{\Delta} (\boldsymbol{u}_k - \boldsymbol{u}_{k-1}) \tag{16}$$

where $\boldsymbol{\Lambda} = \text{diag}(\boldsymbol{\lambda}) > \boldsymbol{0}$ 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, an ad hoc risk cost function is introduced to keep the risk levels present in the system acceptable:

$$\rho(\boldsymbol{x}_k, \boldsymbol{\theta}_k) = \sum_{j=1}^{J} (\mu_1 + \mu_2 \chi_j V_w) e^{-\zeta(||\mathbf{c}_j - \boldsymbol{p}_k|| - r_j)} \tag{17}$$

with $\boldsymbol{\mu} > \boldsymbol{0}$. Moreover, $\chi_j = \max(0, \hat{\boldsymbol{\iota}}_j \cdot \hat{\boldsymbol{\omega}})$ where $\hat{\boldsymbol{\iota}}_j$ is the unit vector from the ship to each obstacle center and $\hat{\boldsymbol{\omega}}$ is the unit wind direction vector. Note that the risk costs are not formulated as explicit constraints to ensure safe distances between the ship and obstacles. Rather, this formulation utilizes violatable risk costs in order to acknowledge that grounding risks may still be evaluated even if they are very high. 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. Lastly, the dot product scales the wind force contribution toward the land 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 is coming from the west, etc. Negative dot products are however set to zero, disregarding favorable winds with respect to perceived risks.

5. IMPLEMENTATION AND SETTINGS

5.1 MPC scheme overview

Pseudo-code for the complete MPC algorithm is presented in Algorithm 1, and a more detailed rundown is as follows: Vectors, matrices, cost functions and custom functions for numerical integration are set up using the CasADi symbolic framework (Andersson et al., 2012). The implemented MPC module then solves one NLP for each

Algorithm 1 MPC algorithm

Result: optimal control input
initialize empty solution
while *not arrived at destination* **do**
  if *any thrusters online* **then**
    construct NLP ← last state
    optimal states and inputs ← solve NLP
    current state ← extract first optimal step
  else
    current state ← simulate drifting ship
  end
  if *risk* > *accepted maximum threshold* **then**
    activate emergency protocol
  else
    solution ← current state and control inputs
    last state ← current state
  end
end

control interval, using the nonlinear optimization package *Interior Point OPTimizer* (IPOPT) and the linear solver *MUltifrontal Massively Parallel sparse direct Solver* (MUMPS). However, if all inputs are constrained to zero, i.e. during the drifting ship or blackout scenarios, an open loop simulation is used instead. Finally, the total risk at each time step is evaluated through the risk cost model from (17). If the risk level rises above a given maximum acceptable threshold, the algorithm is terminated with the assumption that emergency procedures such as remote control or automatic drop of anchor would be applied.

5.2 Initialization and performance factors

Due to the non-convex nature of the control problem, the initial values given to each NLP for solving have a significant impact on its solutions. In this work, the current state and input vectors, the previous solution as well as the internal solver parameters are given to the next NLP to be solved as initial guesses (warm start), as described in Algorithm 1. However, for the first NLP solve, initial guesses are generated by assuming that the ship will follow the path with a velocity equal to the given reference transit speed. The sampling period of $t_\Delta = 30\,\text{s}$ was chosen to have a reasonable balance between the slowest time constant of the ship dynamics and a desired long prediction horizon. Longer sampling periods give faster MPC tuning and testing, but for certain time-sensitive scenarios however, this may be inadequate. It is recommended that this interval should be reduced if used in applications. Similarly, a control horizon of $N = 40$ was chosen, resulting in a total of 570 decision variables for each NLP. This was set by assessing the solving time of the algorithm with respect to the quality of the computed solutions. Increasing the control horizon past this point seemed not to improve the NLP solutions noticeably, and the average solving time of 2 s across all scenarios was deemed appropriate compared to the total prediction horizon of 20 min. If the NLP solver does not converge to a solution within the maximum limit of 300 iterations, the last solution calculated is used. This maximum iteration limit was set such that the maximum solve time during any emergency scenario would be $27\text{s} < t_\Delta$.

5.3 Model parameters and values

The model parameters and MPC settings are given in Table 3, based on the dimensions and onboard systems of a cargo vessel with a maximum surge velocity of ∼15 knots. The rest of the initial and final values are all set to zero. The remaining cost coefficients of $\boldsymbol{\kappa}$, $\boldsymbol{\lambda}$, $\boldsymbol{\delta}$, $\boldsymbol{\mu}$ and ζ are set constant throughout all scenarios, and were empirically determined in order to produce desired trajectories: The path following and speed-related costs were first established to encourage path progress. Next, the input force costs were tweaked to smooth out gradients and reduce power consumption. Finally, appropriate risk costs were estimated by assessing the resulting trajectories during various wind angle and velocity configurations. This ad hoc evaluation approach should however be replaced by more systematic and robust methods for approximating or learning risk coefficient values in future works.

Parameter	Symbol	Value	Unit
Path step size	α_{step}	75	m
Overall ship length	L_{oa}	75	m
Thruster arm lengths	l_x	33	m
Transit reference speed	s_{ref}	2.5	m/s
Frontal projected area	A_{Fw}	110	m^2
Lateral projected area	A_{Lw}	624	m^2
Max thruster force	$f_{\max}$	200	kN
Max thruster azimuth rate	$\omega_{\max}$	2.0	rpm
Viscous damping force surge	X_u	5.0×10^1	kN s/m
Viscous damping force sway	Y_v	2.0×10^2	kN s/m
Viscous damping force yaw	N_r	3.0×10^4	kN s/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

Table 3. Model parameters

5.4 Visualization

Figure 2 illustrates how the slope of the separate risk cost term influences path planning and decision-making. The risk values along the z-axis are given as percentages of the maximum risk cost present at a selected time instant, and the x and y axes are given in meters. The surface plot indicates how the gradient of the obstruction- and disturbance-related terms naturally push the trajectory of the ship (the dashed line) down into the "valley" of lower costs between two grounding obstacle examples. This topography is similar to that of artificial potential fields and is unique for each time step, as well as for

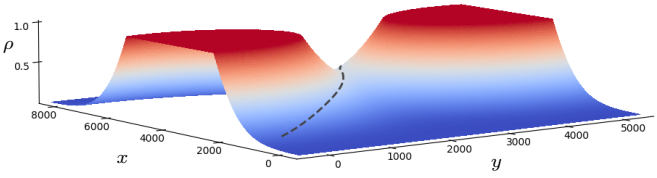

Fig. 2. Surface plot of MPC risk costs at some time instant.

each *predicted* time step into the control horizon N. The risk term surface thus evolves dynamically based on the current ship position and the disturbance circumstances influencing the ship. That is, if the ship moves closer to an obstacle with wind disturbances directed toward it, the slope of the risk cost around that obstacle steepens and rises accordingly. Figure 3 presents the map module used for visualization of the scenario simulations, which displays polygon data of a Norwegian fjord area shown using the Mercator projection. A color bar indicates the ocean depths for the surrounding maritime environment, here used for display purposes only. The preplanned reference path is shown as a dashed gray line. Additionally, red circles mark the grounding obstacles $\boldsymbol{\sigma}$ used by the MPC for cost function calculations. In this work, these obstacle circles are static, and are deliberately made simplistic for proof of concept. Methods for dynamic calculation of grounding areas more shallow than the maximum draft of the ship should however be used in applications. This is left to be further expanded upon in future works.

6. RESULTS

Ceasing normal operations mid-transit is considered a costly action. The main goal of the decision-making algorithm is thus to assess and optimize the risk levels against other operational costs at each time interval, and determine to what degree the ship should follow its initial route within some safety threshold. This section presents the MPC performance of seven scenarios in a simulated maritime environment, to demonstrate the capabilities of the algorithm during extreme circumstances. System stability follows in general from optimality of solutions and by utilizing an adequately long horizon (Grüne and Pannek, 2017), as well as through low-level controllers used by the ship's propulsion subsystems. Figures 4, 5, 6, and 8 show the generated ship trajectories for the investigated scenarios, plotted every 2 min for visibility. Figure 7 shows time series plots of the azimuth thrusters during runtime.

6.1 Impaired thrusters

The defective propulsion system failure mode is split into two separate faults, with no wind disturbances for analysis purposes. Figure 4 shows both scenario simulations in red and yellow, as well as a trajectory during normal

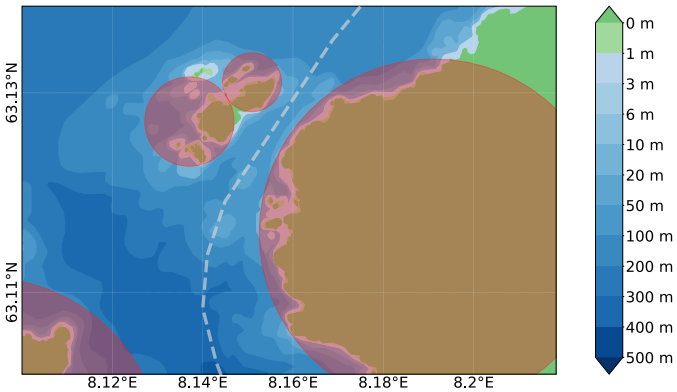

Fig. 3. Visualization of the circular obstacles used by the MPC algorithm during cost function calculations.

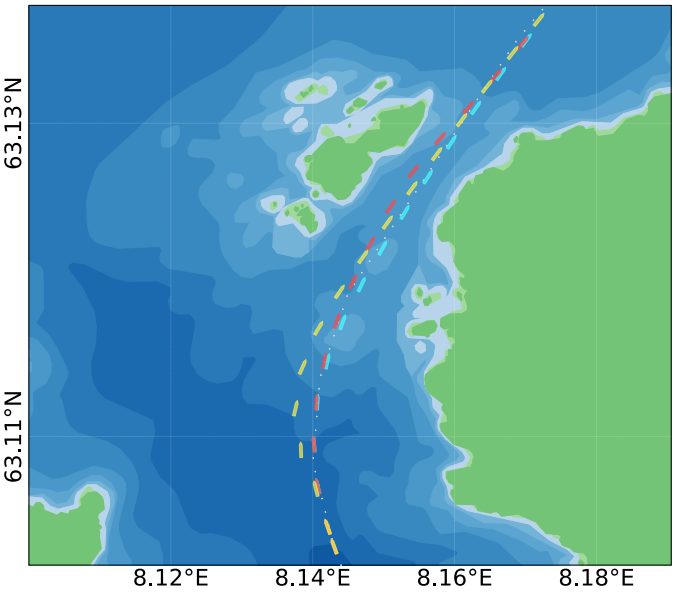

Fig. 4. Simulations during normal operations (cyan), a temporary loss of steering power scenario (red), and a single offline thruster scenario (yellow).

operations in cyan, for reference. It may be noted how the normal operations trajectory deviates slightly from the planned path, due to imposed thruster input costs and the internal prediction horizon each time step. This is a result of the preplanned piecewise reference path being linear and non-optimal. The red ship trajectory depicts the scenario (**1.a**) in which the azimuth thruster rotators are disabled after 5 min e.g. due to an auxiliary system fault, effectively leaving the ship with limited steering capabilities for a time period. However, the algorithm is capable of adjusting the thruster propulsion forces appropriately to continue along an almost identical trajectory until the thruster rotation capabilities are restored, by utilizing the current azimuth angles and varying their propulsion output appropriately. Straying away some distance from the path and closer to the shoreline is considered sufficiently safe, and the ship thus continues ahead despite the imposed system faults. In the second scenario (**1.b**) shown in yellow, the bow thruster shuts down after 3 min, and stays offline until the ship reaches its destination. It is nevertheless apparent that the ship is still able to continue its voyage safely and almost as quickly as the nominal trajectory, even with only the stern thruster available. Note that the algorithm has no access to information about future faults, and consequently is forced to operate with only its current knowledge each control interval. It is considered trivial to investigate less extreme scenarios than the total steering and propulsion loss cases presented here.

6.2 Total blackout

In this failure mode, the ship experiences a power blackout and drifts due to moderate disturbances. An open-loop simulation produces new states each time step during this time period, as there are no available degrees of freedom through the zero-constrained system inputs. If and when to drop the anchor is consequently the only decision left to make in these circumstances. The act of dropping the anchor is assumed to be an exceptionally costly action, and

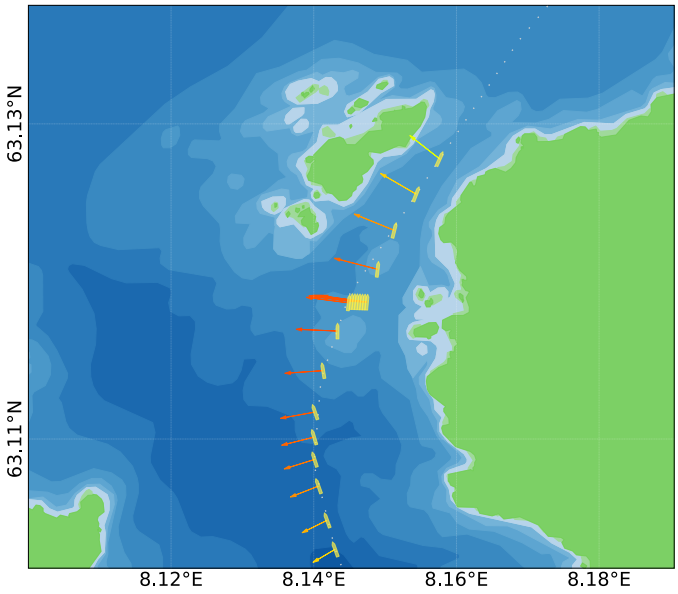

Fig. 5. A temporary blackout and recovery scenario.

is thus postponed as long as reasonably possible in order to give the crew a chance to restart the engines before doing so. Both scenarios have $V_w = 10\,\mathrm{m/s}$ set constant for consistency and proof of concept. Color-coded risk gradient vectors are added to the ship trajectory plots every time step. These vectors show how the perceived risk cost penalizes staying close to the shoreline during difficult weather conditions. The arrows have lengths inversely proportional to $||\boldsymbol{c}_j - \boldsymbol{p}_k|| - r_j$, and are parallel to $-\hat{\boldsymbol{\iota}}_j$. Yellow to red arrow colors represent a medium to large scalar product χ_j, respectively. Figure 5 shows the first scenario (**2.a**) with $\psi_w = 90°$ (east), demonstrating how the ship may still salvage the situation and complete its mission if a blackout is recovered from quickly. Though the ship deviates from the planned path, the solutions of the NLP solver are able to swiftly find their way back to resume the mission as normal. The second scenario (**2.b**)

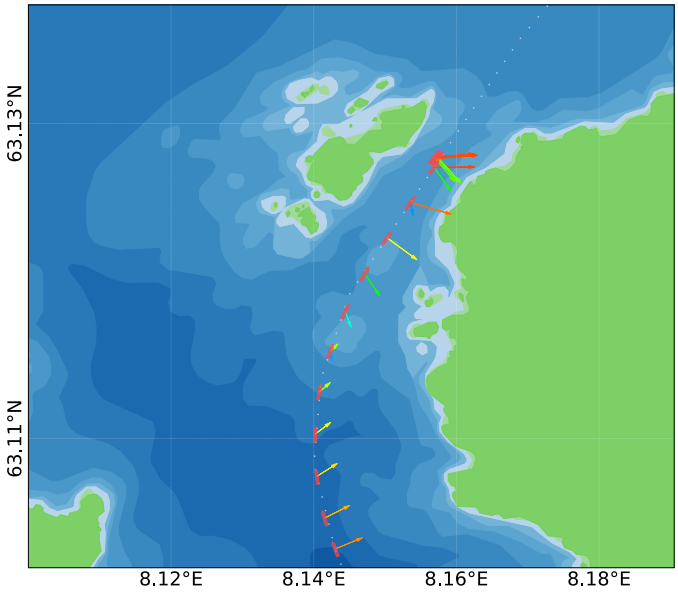

Fig. 6. A blackout too close to the shoreline, leading to an anchor drop.

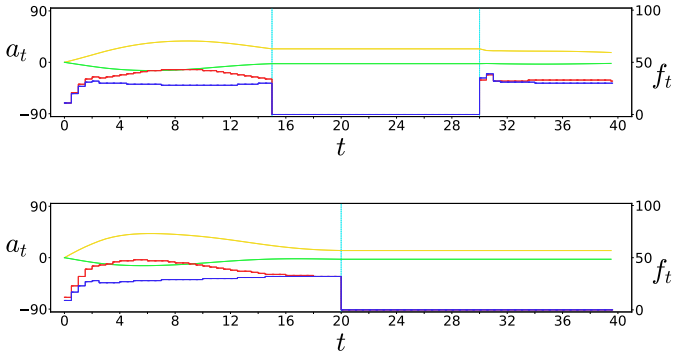

Fig. 7. Thruster inputs for the temporary blackout and blackout with anchor drop scenarios, respectively.

with $\psi_w = -90°$ (west) is shown in Figure 6, in which the ship gets too close to the shoreline after a blackout. Thus, the risk rises above the given threshold, and the anchor is ultimately dropped. Figure 7 contains the time series plots of the azimuth thrusters for each blackout scenario. The time axis is given in minutes, and azimuth angles $\boldsymbol{a}_t$ for the stern and bow thrusters are shown in green and yellow, respectively. The propulsion forces $\boldsymbol{f}_t$ applied by the stern and bow thrusters are respectively shown in blue and red, given as percentages of the maximum thruster force $f_{\max}$.

6.3 Strong winds and risk cost tuning

The choice of specific cost tuning strategies play a critical part in both the trajectory planning algorithm and the anchor drop decision-making. In these hazardous scenarios with strong winds where $V_w = 20\,\mathrm{m/s}$ and $\psi_w = 45°$ (north-east), it is demonstrated in Figure 8 how particular risk cost tuning strategies may lead to different desired behaviors. First, the cyan trajectory (**3.a**) shows a fully completed transit. The risk level does not exceed the maximum threshold, and the algorithm is allowed to continue as normal despite the increased perceived risk levels due to greater wind disturbances. The second trajectory in

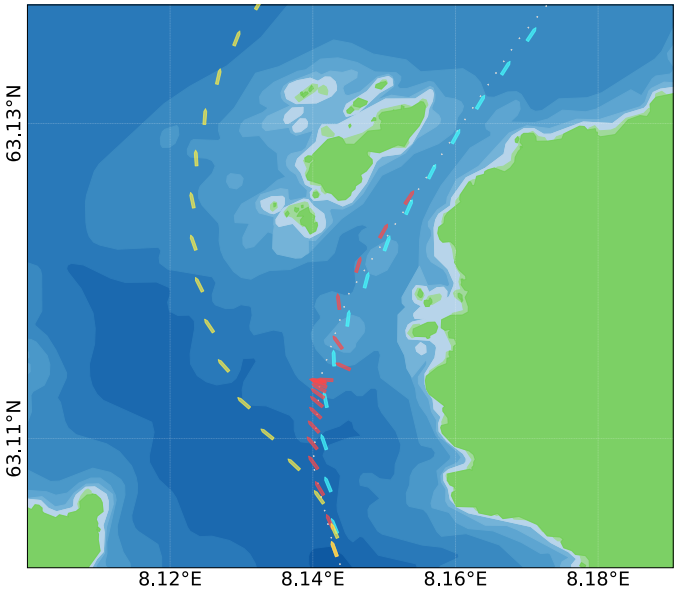

Fig. 8. Different cost tuning approaches generate diverse behaviors during strong winds.

red, however, (**3.b**) shows how the ship alternatively may hold a position for some time and await improved weather conditions. This tuning approach is based on a significantly increased emphasis on safety. Here, the ship is prohibited to pass the narrow strait due to too high predicted risk levels introduced by the strong winds, until the wind velocities recedes after 25 min. This temporary dynamic positioning (DP)-like behavior demonstrates some of the inherent versatility of the MPC scheme, due to the combined steep gradients of the nominal grounding risk and the wind disturbance risk from (17) creating a local minimum (in an undesirable crosswind pose due to initial conditions), blocking the path progression forward. An appropriate system improvement in this situation may be to also consider making an anchor drop if the wait time exceeds some given time period. Alternatively, a second similar approach may be applied, in which the control of the ship is given to a separate DP controller if the opposing risk cost gradients rises above some threshold. The last scenario in yellow (**3.c**) showcases a different cost tuning philosophy, in which the NLP solutions steer the ship around the smaller isles of the narrow strait if the reference path cost coefficient κ_1 is sufficiently relaxed. Which of these approaches is the most suitable for any given scenario may vary greatly depending on specific environmental conditions, ship dimensions and system configurations. These diverging trajectories do however demonstrate another aspect of the flexibility of the MPC algorithm, allowing one to carefully tune cost parameters for several alternative desired behaviors.

7. CONCLUSION

The results from various simulation scenarios show that the MPC algorithm is capable of managing extreme cases of different ship failure modes to a satisfying degree. The implemented system typically solves each nonlinear program with a prediction horizon of 20 min in approximately 2 s, and is thus considered a relatively fast and online decision-making algorithm. There is however much potential for improvement within both the algorithm itself and the complexity of the investigated scenarios, e.g. by combining scenarios to produce more intricate behaviors. Furthermore, it is recommended that the ad hoc risk function implemented for proof of concept should be replaced with probabilistic risk models, and that further advancements of the path progression or risk cost functions may be achieved by adding additional terms related to other operational objectives. This is left to be explored in future works. Ultimately, risk-based MPC is considered an appropriate method for trajectory planning and decision-making for autonomous ships during emergencies.

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 number 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 number 223254), with special thanks to my colleagues Tobias Torben and Thomas Johansen for their inputs on marine dynamical modeling and risk management.

REFERENCES

Anderson, S.J., Peters, S.C., Pilutti, T.E., and Iagnemma, K. (2011). Design and Development of an Optimal-Control-Based Framework for Trajectory Planning, Threat Assessment, and Semi-autonomous Control of Passenger Vehicles in Hazard Avoidance Scenarios. In *Springer Tracts in Advanced Robotics (2011) 70(STAR)*. Springer, Berlin, Heidelberg.

Andersson, J., Åkesson, J., and Diehl, M. (2012). Casadi: A symbolic package for automatic differentiation and optimal control. In *Recent advances in algorithmic differentiation*, 297–307. Springer.

Chen, Y., Peng, H., and Grizzle, J. (2018). Obstacle avoidance for low-speed autonomous vehicles with barrier function. *IEEE Transactions on Control Systems Technology*, 26(1), 194–206.

Eriksen, B.O.H. and Breivik, M. (2017). MPC-based mid-level collision avoidance for ASVs using nonlinear programming. In *2017 IEEE Conference on Control Technology and Applications (CCTA)*, 766–772. IEEE.

Fossen, T.I., Breivik, M., and Skjetne, R. (2003). Line-of-sight path following of underactuated marine craft. In *6th IFAC Conference on Manoeuvring and Control of Marine Craft (MCMC 2003), Girona, Spain*, 244249.

Fossen, T.I. (2011). *Handbook of Marine Craft Hydrodynamics and Motion Control*. John Wiley & Sons.

Grüne, L. and Pannek, J. (2017). Nonlinear model predictive control. In *Nonlinear Model Predictive Control*, 45–69. Springer.

Johansen, T.A., Cristofaro, A., and Perez, T. (2016). Ship collision avoidance using scenario-based model predictive control. *IFAC-PapersOnLine*, 49(23), 14–21.

Keviczky, T., Falcone, P., Borrelli, F., Asgari, J., and Hrovat, D. (2006). Predictive control approach to autonomous vehicle steering. In *2006 American Control Conference*, 6 pp.

Kufoalor, D., Johansen, T.A., Brekke, E.F., Hepsø, A., and Trnka, K. (2019). Autonomous maritime collision avoidance: Field verification of autonomous surface vehicle behavior in challenging scenarios. *Journal of Field Robotics*.

Li, Z. and Sun, J. (2012). Disturbance Compensating Model Predictive Control With Application to Ship Heading Control. *IEEE Transactions on Control Systems Technology*, 20(1), 257–265.

Morrison, D.D., Riley, J.D., and Zancanaro, J.F. (1962). Multiple shooting method for two-point boundary value problems. *Communications of the ACM*, 5(12), 613–614.

Samuelson, S. and Yang, I. (2018). Safety-aware optimal control of stochastic systems using conditional value-at-risk. In *2018 Annual American Control Conference (ACC)*, 6285–6290. IEEE.

Zhou, H., Guvenc, L., and Liu, Z. (2017). Design and evaluation of path following controller based on MPC for autonomous vehicle. In *Chinese Control Conference, CCC*, 9934–9939.