Non-angular MPC-based Thrust Allocation Algorithm for Marine Vessels
- A Study of Optimal Thruster Commands

Stian Skjong, Eilif Pedersen

***Abstract*—In this work a thrust allocation algorithm for marine vessels based on model predictive control theory (MPC) and a non-angular vector formulation is presented and studied. The main objective in this work is to highlight the potentials of using an optimal thrust allocation algorithm including a time horizon to reduce the power consumption as well as reducing the environmental disturbances in the thruster commands. The proposed thrust allocation algorithm is compared to a one-step optimization algorithm in a benchmarking test. A one-step thrust allocation algorithm is an optimization algorithm with a time horizon that includes only one sample. When using a longer time horizon in the proposed algorithm the thrust allocation has the potential of optimizing rate limited states in the long run, e.g. whether it would be beneficial to rotate a thruster or to increase or decrease the commanded thrust when thruster biasing is considered as an option. Preliminary case studies are presented where different cost function weights and horizon lengths are compared. The finite time horizon in the MPC thrust allocation algorithm also makes it possible to affect the dynamics of the optimized thruster signals since it can use the entire time horizon to reach its objective. This is very important when considering reducing the thrust rates when controlling a marine vessel in Dynamic Positioning- (DP) operations since wave-filters never succeed in filtering out all oscillatory environmental effects. Thus, an optimal thrust allocation algorithm with well chosen cost function weights, along with thruster biasing, would reduce the magnified oscillations in the produced thrust, while keeping the power consumption at a minimum, which has been devoted the main focus in this work.**

***Index Terms*—Thrust Allocation, Model Predictive Control, Optimization, Optimization Horizon, Sampled Systems, Reduced thrust oscillations**

I. Introduction

In marine offshore operations, proper control of marine vessels and equipment are important both in order to complete the task at hand at the same time as keeping the costs at a minimum and maintaining a financial surplus. This must be done within the requirements set by the customers in order to maintain a good reputation such that new contracts can be made easier. On the other hand, marine operations tend to become more demanding, requiring higher precisions and special qualifications at the same time as environmental footprints have become more highlighted than before. Since every third party vendor put a lot of effort into research and development of their products, the largest potential of improvements of a vessel in specific operations is within system integration, namely how the vessel's equipment should be interfaced and controlled in order to obtain the best performance possible. One such important system integrator is the thrust allocation algorithm, which connects and transforms the global commanded thrust signals from the Dynamic Positioning (DP)-controller to the propulsion system.

A lot of work has been done regarding thrust allocation for marine vessels, as well as in the aerospace industry. It has been growing interests in formulating the thrust allocation problems as Model Predictive Control (MPC) problems [1], thus, most often including thruster angles and absolute thrust in the problem formulation. In general, an MPC is an optimization based method for the feedback control of a system. Model predictive control is also known as a moving horizon control (MHC) and receding horizon control (RHC) because it optimizes over a given time horizon, as will be elaborated in section II, and is often used for controlling slow dynamical systems [2]. A thorough survey of model predictive control theory and practice is given in [3], and in [4] the recent developments and future promises of MPC is discussed. Hence, these topics will not be given much attention in this work. For a thorough introduction to both linear and non-linear MPC theory the reader is referred to [5] and [6], respectively.

A thrust allocation method with dynamic power consumption modulation for diesel-electric ships is presented in [7]. This algorithm leads to a more stable loading of the power plant for reduction in fuel consumption, in addition to reducing wear of the power plant, in comparison to standard thrust allocation algorithms. In [8] a robust control allocation for over-actuated ships is considered and verified by experiments with a model ship. Much attention is devoted to reducing the load variations on the power plant through proper thrust allocation designs using MPC in [9], where a thrust allocation algorithm including fuel consumption characteristics is presented, and [10], where the thrust allocation algorithm includes power management functionalities for reduction in frequency and load variations on the electric network. Also, in the field of aerospace, control allocation problems have been addressed in [11], [12], and are similar to the thrust allocation problem in the marine environment.

In general, the thrust allocation problem for a marine vessel can be solved explicitly [13] for non-rotatable actuators, as done in [14] and [15]. However, when considering rotatable thrusters, the thrust allocation problem becomes implicit, and

S. Skjong is a Ph.D candidate at the Department of Marine Technology, Norwegian University of Science and Technology (NTNU), e-mail: stian.skjong@ntnu.no.

E. Pedersen is an Assiciate professor at the Department of Marine Technology, Norwegian University of Science and Technology (NTNU), e-mail: eilif.pedersen@ntnu.no

is often not convex when only allowing thruster angles to have numerical values within certain regions [16]. However, in [15] and [17] an explicit method for solving the thrust allocation problem using rotatable thrusters and piecewise linear functions is proposed and can be used in order to include rotatable thrusters in one-step optimization problems as well. In industrial thrust allocation algorithms, the use of one-step optimization is prevalent. This, because of its simplicity as well as being generally fast to solve, which is an important requirement in real-time implementations.

In some applications the DP-controller is integrated into the thrust allocation optimization problem, as done in [18], in contrast to a stand-alone algorithm feeding the thrust allocation algorithm with global thrust commands. Also, the fidelity of DP-controllers span from model based control designs [19] and DP-controllers including advanced filters [20], [21] to simple PID-control based designs including rotational matrices [22]. A thorough survey of DP-control systems is given in [23].

In the closed control loop, consisting of the available measurements, the DP-controller, the thrust allocation algorithm and the propulsion system, there is often a filter as well. This filter takes the available position and orientation measurements and tries to filter out the fastest oscillatory environmental effects. Such filters could be based on system dynamics such as Kalman filters [24] and non-linear passive observers [25], that also include observer properties, or simple filters such as band-pass, band-stop or low-pass filters. Although suited filters are used, it is hard to filter out all the unwanted wave- and environmental effects without introducing a significant phase-lag, and these effects tend to be amplified through the DP-control system if real care is not taken. Hence, filtering properties in the thrust allocation algorithm are much appreciated.

In this work a non-angular MPC-based thrust allocation algorithm framework for maritime surface vessels in DP-operations, including an optimization horizon larger than the one-step method, is proposed, tested and benchmarked against a standard non-linear one-step thrust allocation algorithm. By including an optimization horizon of proper length it will be shown that the proposed thrust allocation algorithm can obtain filtering properties that not only reduce oscillatory environmental disturbances, but also maintain a low power consumption while keeping the vessel in position. Because of the new optimization problem formalism presented in this work, that excludes thruster angles in the problem formulation, the algorithm would also have good real-time properties even though having a significant optimization horizon length that increases the problem quadratically, as will be shown in various simulation results presented in this article. In contrast to the algorithms presented in the literature, the proposed algorithm is simple to formulate while not being subject to piecewise linearisation, multiple shooting strategies or additional functionalities for ensuring faster convergence, which saves development time.

The proposed thrust allocation algorithm does not contain any information about fuel consumption in the power plant, in contrast to the already mentioned literatures. However, it takes aim at reducing the thrust commands and the thrust command oscillations in order to both smooth the power consumption as well as keeping the loading of the power plant as low as possible. This, because the consumed power is approximately proportional to the generated thrust to the power of $\frac{3}{2}$, see [18].

The main case study presented in this work, which includes a high fidelity vessel model with all relevant auxiliary systems as the one presented in [26], provides reasonable and realistic results which is crucial when optimizing system integration such as the interaction between the measurements, the filter, the DP-controller, the thrust allocation algorithm and the local thruster controllers, as done in this work, where the thrust allocation algorithm is the main integrator between the different systems. Different key performance indicators (KPI's) such as total power consumption, position and orientation errors, and power spectral density (PSD) analysis of the thrust command outputs, are used to compare different sets of cost function weights in the thrust allocation algorithm. This type of thorough study is not very prevalent in the literature. Hence, it has been devoted much attention in this work.

When working with non-linear MPC problems, often additional work must be put into assuring global convergence of the optimization problem, such as adding soft constraints [27]. However, such soft constraints are not needed in the proposed algorithm. A non-angular vector formulation contains the same amount of information as an angle-amplitude formulation, but the drawbacks are that the thrust angles and the thrust amplitudes must be calculated from the vectors before being fed to the propulsion system, and that the thrust rates and the thrust constraints tend to become a bit conservative. However, as will be shown in section II-D, calculating the thrust angles and amplitudes are a simple and explicit procedure.

The reason for using an MPC-algorithm instead of a one-step optimization algorithm is primarily to be able to work with thrust rates since such algorithms often runs around 1 Hz. Then, it is possible to integrate the optimal rates outside the thrust allocation algorithm in order to obtain smooth thrust commands. When using a one-step algorithm, this is much more difficult since the algorithm has only one sample in the horizon before reaching the reference values, which compromises reducing the rates. A longer optimization horizon also enables planning of the thrust commands on a future time horizon, in contrast to a one-step algorithm. This effect is studied in a benchmarking test that compares the proposed algorithm and a one-step algorithm in a simple DP-operation case.

A. Structure of Article

This article is structured as follows: First, the proposed thrust allocation algorithm is presented in section II before being benchmarked against a one-step thrust allocation algorithm in a DP-operation simulation case in section III. Secondly, simplified preliminary case studies including only the thrust allocation algorithm are initiated in section IV in order to highlight how the characteristics of the proposed thrust allocation algorithm are affected by different cost function weights and optimization horizon lengths. In the end, a main case study is presented in section V and shows how different choices of cost

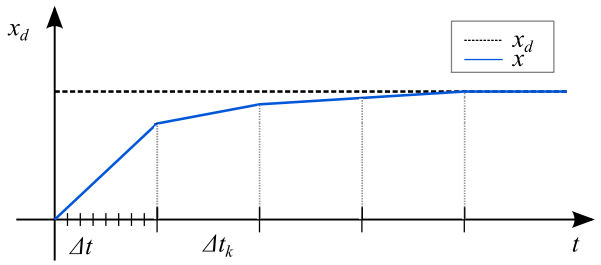

Fig. 1. Sketch of solution from the thrust allocation algorithm where x_d is the desired thrust and x is the corresponding output from the thrust allocation algorithm

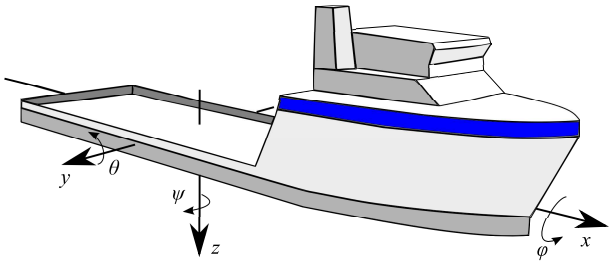

Fig. 3. Body-fixed reference frame of the vessel

function weights affect the overall performance of a vessel in DP-operation excited by current and irregular sea states.

II. MPC-based Thrust Allocation Algorithm

In general, solving optimization problems, such as MPC problems, often tend to become computationally demanding and the optimization problem is often implemented with discrete dynamics and with a fixed number of samplings in the horizon. Here, the number of steps in the horizon, K, is given as

$$K = \text{floor}\left(\frac{T}{\Delta t_k}\right) \tag{1}$$

where T is the length of the time horizon treated in the optimization and Δt_k is the length of each sample in the horizon, as well as the time step between each optimization solver call. Hence, the thrust optimization is performed with a frequency $f_k = \Delta t_k^{-1}$, outputting thrust rates that can be integrated in between each optimization with a time step Δt. Fig. 1 gives an overview of how these time steps relate to each other.

Before presenting the proposed thrust allocation algorithm, a few definitions are needed. A thrust-vector representation is to be used instead of the traditional amplitude-angle representation for rotatable thrusters in order to reduce non-linearities in the optimization problem related to the thruster angles. This means that each rotatable thruster can be represented by two thrust vectors, F_x and F_y, as shown in Fig. 2. These two thrust vectors include the same amount of information as the traditional amplitude-angle representation, given as F_c and α, respectively, in the figure. Relations between the amplitude-angle based thrust representation and the vector representation are given as

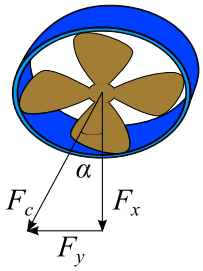

Fig. 2. Thrust from propulsor given as both vector representation and amplitude-angle representation

$$F_c = \sqrt{F_x^2 + F_y^2} \tag{2a}$$

$$\alpha = \arctan\left(\frac{F_y}{F_x}\right) \tag{2b}$$

Also, the sign convention for the thrust vectors are set according to the body-fixed reference frame of the vessel, as shown in Fig. 3. This means that a positive thrust in F_x moves the ship forward and a positive thrust in F_y moves the ship to starboard. For a thruster with fixed azimuth angle, such as a bow thruster, the thrust amplitude is used along with the static thrust angle, α_s.

When having N number of thrusters, the global thrust vectors in surge, sway and yaw, denoted as $X_N(t_k)$, $Y_N(t_k)$ and $Mz_N(t_k)$, respectively, for time step t_k, can be calculated as

$$X_N(t_k) = \sum_{i=1}^{N} F_{ix}(t_k) \tag{3a}$$

$$Y_N(t_k) = \sum_{i=1}^{N} F_{iy}(t_k) \tag{3b}$$

$$Mz_N(t_k) = -\sum_{i=1}^{N} \left[F_{ix}(t_k)y_i - F_{iy}(t_k)x_i\right] \tag{3c}$$

where (x_i, y_i, z_i) is the position of thruster i. For convenience, these total thrust contributions are given in vector form for time step t_k as

$$\boldsymbol{x}(t_k) = \left[\ X_N(t_k),\quad Y_N(t_k),\quad Mz_N(t_k)\ \right]^\top \tag{4}$$

and the corresponding desired thrust given by a potential DP-controller is given as $\boldsymbol{x}_d(t_k)$. Note that the thrust command is assumed constant during the whole horizon. This can be argued for when having a DP-controller that provides the desired thrust command, containing integration effects, in combination with a small Δt_k. However, this assumption should be verified through simulations.

A bounded variable f is said to be defined in the range $f \in [\underline{f}, \overline{f}]$ such that

$$\begin{aligned} \min(f) &= \underline{f} \\ \max(f) &= \overline{f} \end{aligned} \tag{5}$$

Also, the bound vector is defined as

$$\overline{\underline{f}} = [\underline{f}, \overline{f}] \tag{6}$$

An absolute value, F_c, of two thrust vectors F_x and F_y, as in (2a), is said to be signed if it is negative when the thruster is reversed and positive if not, and is denoted as $F_c^{\pm}$. Also note that $k \in [1, .., K]$ is the sampling number of the MPC horizon and is used as a place-holder for the time step k, t_k, such that $[t_0, t_1, .., t_k, .., t_K]$ and $t_K = T$.

In the thrust allocation algorithm presented in this work, physical limitations need to be addressed in the algorithm. This is done through constraints in the optimization problem, and is elaborated in the following.

A. Constraints

In an optimization problem, constraints are used in order to assure that the optimal solution is realistic and possible to implement in a realistic system. Often these physical constraints are related to maximal values and rates, e.g. taking into consideration the maximal capacity of the system and how fast the system states are able to change.

Here, all constraints will be presented in continuous time even though being implemented in discrete time in section II-C. The thrust vectors are assumed to be represented as states in the thrust allocation algorithm in order to include rate-limitations and to be able to calculate new thrust commands in between each optimization call, Δt_k. Hence, the thrust vectors are expressed as

$$\frac{d}{dt}F_x(t) = u_x(t) \tag{7a}$$
$$\frac{d}{dt}F_y(t) = u_y(t) \tag{7b}$$

for a rotatable thruster, where u_x and u_y are control variables. Note that these differential equations are not comparable to the differential equations describing a thruster, it is only an implementation for enabling rate limitations for the thrust commands. If the thruster angle is fixed, only one control variable is needed, u, and u_x and u_y can be calculated from u and the static thruster angle α_s. Then, the thrust rate constraints for thruster i can be set as

$$\underline{u_{ix}} \leq u_{ix}(t) \leq \overline{u_{ix}} \tag{8a}$$
$$\underline{u_{iy}} \leq u_{iy}(t) \leq \overline{u_{iy}} \tag{8b}$$

where $[\underline{u_{ix}}, \overline{u_{ix}}]$ and $[\underline{u_{iy}}, \overline{u_{iy}}]$ are the allowed regions for the thrust vector rates. For rotatable thrusters the allowed regions can be calculated in between each optimization, or simply assumed given as

$$\overline{\underline{u_{ix}}} = \overline{\underline{u_{iy}}} = \frac{1}{\sqrt{2}}\overline{u_{ic}} \tag{9}$$

and for fixed thrusters

$$\overline{\underline{u_{ix}}} = \overline{\underline{u_{ic}}}\cos(\alpha_{is}) \tag{10a}$$
$$\overline{\underline{u_{iy}}} = \overline{\underline{u_{ic}}}\sin(\alpha_{is}) \tag{10b}$$

where $[\underline{u_{ic}}, \overline{u_{ic}}]$ is the allowed thrust rate region for F_{ic}. Similarly, the maximal capacity constraint for thruster i can be formulated as

$$\underline{F_{ix}} \leq F_{ix}(t) \leq \overline{F_{ix}} \tag{11a}$$
$$\underline{F_{iy}} \leq F_{iy}(t) \leq \overline{F_{iy}} \tag{11b}$$

where $[\underline{F_{ix}}, \overline{F_{ix}}]$ and $[\underline{F_{iy}}, \overline{F_{iy}}]$ are the allowed thrust vector regions in each direction. As for the thrust rates, the allowed thrust regions for rotatable thrusters can be calculated in between each optimization, or simply assumed given as

$$\overline{\underline{F_{ix}}} = \overline{\underline{F_{iy}}} = \frac{1}{\sqrt{2}}\overline{\underline{F_{ic}}} \tag{12}$$

and for fixed thrusters

$$\overline{\underline{F_{ix}}} = \overline{\underline{F_{ic}}}\cos(\alpha_{is}) \tag{13a}$$
$$\overline{\underline{F_{iy}}} = \overline{\underline{F_{ic}}}\sin(\alpha_{is}) \tag{13b}$$

where $\overline{\underline{F_{ic}}} = [\underline{F_{ic}}, \overline{F_{ic}}]$ is the allowed thrust region for F_{ic}. Note that the rate constraints and the maximal capacity constraints in (9) and (12), respectively, are a bit conservative for the rotatable thrusters. However, as will be seen in section V, much focus will be given to reducing power consumption and keeping the thruster rates as low as possible to avoid large accelerations. Then, conservative rate constraints will have a minor impact on the optimal solution.

When adding the turning rate constraint on each rotatable thruster, e.g. $\frac{d}{dt}\alpha_i$, the orientation and the rate of the thrusters are not directly part of the general vector representation. However, the turning rate for thruster i can be calculated by differentiating (2b). Thus,

$$\begin{aligned}\dot{\alpha}_i(t) &= \frac{d}{dt}\arctan\left(\frac{F_{iy}(t)}{F_{ix}(t)}\right) \\ &= \frac{u_{iy}(t)F_{ix}(t) - u_{ix}(t)F_{iy}(t)}{F_{ix}(t)^2 + F_{iy}(t)^2}\end{aligned} \tag{14}$$

By defining the allowed angle rate region as $[\underline{\dot{\alpha}_i}, \overline{\dot{\alpha}_i}]$ for a thruster i, the rate constraints can be expressed as

$$\begin{aligned}&u_{iy}(t)F_{ix}(t) - u_{ix}(t)F_{iy}(t) \\ &\leq \overline{\dot{\alpha}_i}(t)(F_{ix}(t)^2 + F_{iy}(t)^2),\end{aligned} \tag{15a}$$

$$\begin{aligned}&\underline{\dot{\alpha}_i}(t)(F_{ix}(t)^2 + F_{iy}(t)^2) \\ &\leq u_{iy}(t)F_{ix}(t) - u_{ix}(t)F_{iy}(t)\end{aligned} \tag{15b}$$

The last constraints to be added are the initial conditions to the differential states, namely the thrust vectors. Hence, for a thruster i the initial conditions are given as constraints as

$$F_{ix}(0) := F_{ix,0} \tag{16a}$$
$$F_{iy}(0) := F_{iy,0} \tag{16b}$$

where $F_{ix,0}$ and $F_{iy,0}$ are either the previously obtained states or measurements. Along with constraints, cost functions are added to reflect the chosen optimal thrust. These cost functions are elaborated in the following.

B. Cost Functions

For such optimization problems as MPC's, it would be beneficial for the solving procedure if the cost functions are convex. This can be obtained by using quadratic cost functions, e.g. $\boldsymbol{z}^\top \boldsymbol{Q}\boldsymbol{z}$, where $\boldsymbol{z} \in \mathcal{R}^N$ is a vector and $\boldsymbol{Q} \in \mathcal{R}^{N\times N}$ is a weighting matrix, typically diagonal. Also, in some cases linear costs are added in order to improve convergence when the variables in the cost functions become small. However, this has not been deemed necessary here.

Since a time horizon with a fixed number of samplings is used in the thrust allocation algorithm, two sets of cost functions should be used; one intermediate cost function and one end cost function. Typically, the end cost function would represent the costs from the infinite time horizon that are

Fig. 4. Biasing of main thrusters placed at the stern

neglected when assuming a finite time horizon, while the intermediate cost function, in addition to representing the main objective, would include costs that affect how the main objective is obtained.

Starting with the main objective, the quadratic intermediate cost function for time step k can be expressed as

$$[\boldsymbol{x}(k) - \boldsymbol{x}_d(k)]^\top \boldsymbol{Q}_k [\boldsymbol{x}(k) - \boldsymbol{x}_d(k)] \tag{17}$$

where k is the sampling number, $\boldsymbol{Q}_k \in \mathcal{R}^{3\times3}$ is a diagonal weight matrix and $\boldsymbol{x}(k), \boldsymbol{x}_d(k) \in \mathcal{R}^3$ are the global thrust vector and global demanded thrust vector, respectively. Similarly, the end cost function can be expressed as

$$[\boldsymbol{x}(T) - \boldsymbol{x}_d(T)]^\top \boldsymbol{Q}_T [\boldsymbol{x}(T) - \boldsymbol{x}_d(T)] \tag{18}$$

where $\boldsymbol{Q}_T \in \mathcal{R}^{3\times3}$. Rate costs are assumed negligible in the end cost function since all large accelerations and rates are largest in the first samples in the time horizon and the thrust commands from the DP-controller are assumed constant during the whole time horizon.

In addition to the main objective cost function, rate costs and magnitude costs are added to the thrust control variables and the thrust vectors, respectively. Hence, the rate costs for N number of thrusters for time step k are given as

$$\boldsymbol{u}(k)^\top \boldsymbol{Q}_u \boldsymbol{u}(k) \tag{19}$$

where $\boldsymbol{u}(k) \in \mathcal{R}^{2N}$,

$$\boldsymbol{u}(k) = \begin{bmatrix} u_{1x}(k), & u_{1y}(k), & ..., & u_{Nx}(k), & u_{Ny}(k) \end{bmatrix}^\top \tag{20}$$

and $\boldsymbol{Q}_u \in \mathcal{R}^{2N\times2N}$ is a diagonal weight matrix. The magnitude costs are given similarly,

$$\boldsymbol{F}(k)^\top \boldsymbol{Q}_F \boldsymbol{F}(k) \tag{21}$$

where $\boldsymbol{F}(k) \in \mathcal{R}^{2N}$,

$$\boldsymbol{F}(k) = \begin{bmatrix} F_{1x}(k), & F_{1y}(k), & ..., & F_{Nx}(k), & F_{Ny}(k) \end{bmatrix}^\top \tag{22}$$

and $\boldsymbol{Q}_F \in \mathcal{R}^{2N\times2N}$. In addition, if the ship has two main thrusters placed at the stern, it would in some cases be of interest to cancel them against each other or to store available thrust in order to obtain a faster response, as shown in Fig 4, and is often referred to as thruster biasing. Note that thruster biasing is also used in some applications for singularity avoidance, but this is not the case here. Thruster biasing can be included in the optimization problem by changing the cost function given in (21) to

$$[\boldsymbol{F}(k) - \boldsymbol{\delta}(X_d(k))]^\top \boldsymbol{Q}_F [\boldsymbol{F}(k) - \boldsymbol{\delta}(X_d(k))] \tag{23}$$

where $\boldsymbol{\delta}(X_d(k)) \in \mathcal{R}^{2N}$,

$$\boldsymbol{\delta}(X_d(k)) = \begin{bmatrix} 0, & \delta_{1y}(X_d(k)), & ..., & 0, & \delta_{Ny}(X_d(k)) \end{bmatrix}^\top \tag{24}$$

and

$$\delta_{iy}(X_d(k)) = |X_d(k)| r_i \arctan(\alpha_{ib}) \tag{25}$$

for the two main thrusters placed at the stern, where α_{ib} is the biasing angle for thruster i and r_i is the fraction of the total maximal thrust in surge thruster i can provide.

The last intermediate cost function to be added is perhaps the most important one after the main objective, namely a cost function that makes thruster biasing optimal. By adding such a cost function to the optimization problem, the total thrust allocation algorithm has the ability to obtain the main objective without accelerating or de-accelerating the thrusters too fast. This would also benefit the goal of obtaining a smooth power demand by the thrusters. This cost function is given as

$$\boldsymbol{u}(k)^\top \boldsymbol{b}(\boldsymbol{F}(k))^\top \boldsymbol{Q}_{uF} \boldsymbol{b}(\boldsymbol{F}(k)) \boldsymbol{u}(k) \tag{26}$$

where $\boldsymbol{b}(\boldsymbol{F}(k)) \in \mathcal{R}^{2N\times2N}$,

$$\boldsymbol{b}(\boldsymbol{F}(k)) = \begin{bmatrix} \frac{F_{1x}(k)}{F_{1c}(k)+\epsilon} & 0 & \cdots & 0 \\ 0 & \frac{F_{1y}(k)}{F_{1c}(k)+\epsilon} & \cdots & 0 \\ \vdots & \vdots & \ddots & 0 \\ 0 & 0 & 0 & \frac{F_{Ny}(k)}{F_{Nc}(k)+\epsilon} \end{bmatrix} \tag{27}$$

and ϵ is a small number added in order to avoid dividing by zero and $F_{ic}(k)$ is given similarly as in (2a),

$$F_{ic}(k) = \sqrt{F_{ix}(k)^2 + F_{iy}(k)^2} \tag{28}$$

One could perhaps argue for the fact that the cost function in (19) and and the cost function in (26) are similar and can be combined into one cost function. However, since the diagonal terms in (27) scale the cost function weights, and since the cost for the thrust rate u_{ip} is zero when $F_{ip} = 0$, where $p \in (x, y)$, the diagonal terms in (27) must then be updated to $c_i + \frac{F_{ip}}{F_{ic}+\epsilon}$ where $c_i > 0$ is a scaling parameter reflecting the difference between $\boldsymbol{Q}_u(i,i)$ and $\boldsymbol{Q}_{uF}(i,i)$. Hence, since there does not exist any computational advantages of combining (19) and (26), they are kept separated.

It is also possible to add cost functions for minimizing thruster-thruster interactions for thrusters placed side by side, meaning that if one thruster is facing the other it will reduce the efficiency of that thruster since its wake will affect the other thruster. This could for example be included as a function of $\left|\frac{F_{iy}}{F_{ix}}\right|$, since when F_{iy} becomes large in comparison to F_{ix}, the thruster angle is approaching a sway oriented direction. However, such cost functions are not included here.

The total MPC formulation of the thrust allocation algorithm is summarized and put together in the following.

C. Total MPC Formulation

The constraints and cost functions have been presented in section II-A and II-B, respectively. The constraints have been derived in continuous time and the cost functions have been given for given time steps k, except for the end cost function in

(18). By combining all the constraints and the cost functions, the total MPC formulation of the thrust allocation problem can be formulated as

$$\min_{\boldsymbol{u}\in\mathcal{R}} \; J(\boldsymbol{x}, \boldsymbol{x}_d, \boldsymbol{u}, \boldsymbol{F}, \boldsymbol{\delta}(X_d), T) \tag{29a}$$

subject to $\forall i \in [1, ..., N]$ and $\forall k \in [1, ..., K]$

$$F_{ix}(0) := F_{ix,0} \tag{29b}$$
$$F_{iy}(0) := F_{iy,0} \tag{29c}$$
$$F_{ix}(k) := F_{ix}(k-1) + u_{ix}(k)\Delta t_k \tag{29d}$$
$$F_{iy}(k) := F_{iy}(k-1) + u_{iy}(k)\Delta t_k \tag{29e}$$
$$\overline{\boldsymbol{g}_i(k)} \leq \mathbf{0} \tag{29f}$$
$$-\underline{\boldsymbol{g}_i(k)} \leq \mathbf{0} \tag{29g}$$

where the total cost function $J(\cdot)$ is given as

$$\begin{aligned} J(\cdot) = &\sum_{k=1}^{T-1} \left[\boldsymbol{x}(k) - \boldsymbol{x}_d(k)\right]^\top \boldsymbol{Q}_k \left[\boldsymbol{x}(k) - \boldsymbol{x}_d(k)\right] \\ &+ \sum_{k=1}^{T-1} \boldsymbol{u}(k)^\top \boldsymbol{Q}_u \boldsymbol{u}(k) \\ &+ \sum_{k=1}^{T-1} [\boldsymbol{F}(k) - \boldsymbol{\delta}(X_d(k))]^\top \boldsymbol{Q}_F [\boldsymbol{F}(k) - \boldsymbol{\delta}(X_d(k))] \\ &+ \sum_{k=1}^{T-1} \boldsymbol{u}(k)^\top \boldsymbol{b}(\boldsymbol{F}(k))^\top \boldsymbol{Q}_{uF} \boldsymbol{b}(\boldsymbol{F}(k)) \boldsymbol{u}(k) \\ &+ \left[\boldsymbol{x}(T) - \boldsymbol{x}_d(T)\right]^\top \boldsymbol{Q}_T \left[\boldsymbol{x}(T) - \boldsymbol{x}_d(T)\right] \end{aligned} \tag{30}$$

and the inequality constraint vector function $\overline{\underline{\boldsymbol{g}_i(k)}}$ is given as

$$\overline{\underline{\boldsymbol{g}_i(k)}} = \begin{bmatrix} u_{ix}(k) - \overline{u_{ix}} \\ u_{iy}(k) - \overline{u_{iy}} \\ F_{ix}(k) - \overline{F_{ix}} \\ F_{iy}(k) - \overline{F_{iy}} \\ u_{iy}(k)F_{ix}(k) - u_{ix}(k)\overline{F_{iy}}(k) - \underline{\dot{\alpha}_i} F_{ic}(k) \end{bmatrix} \tag{31}$$

and $\Delta t_k = t(k) - t(k-1)$ is the time between two sampling intervals. Note that (29d) and (29e) are the discrete implementations of (7a) and (7b), respectively. Also note that (29g) is a restriction making sure that the lower bound of $\boldsymbol{g}_i(k)$ is larger than zero.

The outputs from the thrust allocation algorithm are $F_{ic}^{\pm}$ and α_i for rotatable thrusters and F_{ic} for fixed ones such as tunnel thrusters. Since the thruster angles for the rotatable thrusters are not directly included in the optimization problem formulation, the angles, α_i, need to be calculated afterwards from the optimal MPC output. The same goes for the signed thrust amplitudes $F_{ic}^{\pm}$. These calculations are elaborated in the following.

D. Implementation

Usually, thrusters are controlled based on thruster angles and signed thrust amplitudes. When using a thrust vector representation to describe the thrust forces generated by each thruster instead of thruster angles and signed thrust amplitudes, some logics must be implemented in order to obtain the desired control signals for the thrusters after the optimization algorithm. In general, logics must be implemented in order to count the number of rotations a thruster goes through, in order to produce a continuous thruster angle signal, and logics that determine whether the MPC rotates or reverses a given thruster.

By using $\arctan 2(\cdot)$ instead of $\arctan(\cdot)$ in (2b), one can count the number of rotations by comparing the previously calculated thruster angle with the current one. This, in order to assure that the angle commands do not contain discontinuities. This procedure is summarized in Algorithm 1.

Algorithm 1 Counting thruster rotations

procedure ROTATIONCOUNT($\cdot$)
  $\alpha_i(t) = 2\pi n_i + \pi m_i + \arctan 2(F_{iy}(t), F_{ix}(t))$
  if $|\alpha_i(t) - \alpha_i(t-1)| \geq 2\pi$ **then**
    if $\alpha_i(t) - \alpha_i(t-1) < 0$ **then**
      $n_i = n_i + 1$
    else
      $n_i = n_i - 1$
    $\alpha_i(t) = 2\pi n_i + \pi m_i + \arctan 2(F_{iy}(t), F_{ix}(t))$
  return $\alpha_i(t)$

In the algorithm, n_i is the rotation counter and m_i is another counter used for calculating whether the thrust allocation algorithm rotates a thruster or reverses the corresponding thrust. This can be evaluated after running Algorithm 1 since the thruster angle signal doesn't contain any discontinuities related to the trigonometric function. Hence, the logics needed to determine whether a thruster is rotated or reversed by the MPC may be implemented as in Algorithm 2.

Algorithm 2 Rotating v.s. reversing thruster

procedure ROTATEORREVERSE($\cdot$)
  $\alpha_i(t) = 2\pi n_i + \pi m_i + \arctan 2(F_{iy}(t), F_{ix}(t))$
  $F_{ic}^{\pm}(t) = s_i\sqrt{F_{ix}(t)^2 + F_{iy}(t)^2}$
  if $|\alpha_i(t) - \alpha_i(t-1)| > \dot{\alpha}_{i,max}\Delta t$ **and** $|F_{ic}^{\pm}(t)| \leq \epsilon$ **then**
    if $|\alpha_i(t)| - |\alpha_i(t-1)| < 0$ **then**
      $m_i = m_i + 1$
    else
      $m_i = m_i - 1$
    $s_i = -s_i$
    $\alpha_i(t) = 2\pi n_i + \pi m_i + \arctan 2(F_{iy}(t), F_{ix}(t))$
    $F_{ic}^{\pm}(t) = s_i\sqrt{F_{ix}(t)^2 + F_{iy}(t)^2}$
  return $\alpha_i(t), F_{ic}^{\pm}(t)$

In the algorithm, s_i is a sign variable, $s_i \in [-1, 1]$, and ϵ is a small number, $\epsilon > 0$. In general, this algorithm checks if the rate constraint for the thruster angle is violated, and if the thrust magnitude is small, then the thruster has been reversed according to the optimization algorithm.

The total thrust allocation algorithm including Algorithm 1 and 2 has been implemented in the C++ library ACADO [28] and solved by the qpoases library [29].

Even if the constraints and the cost functions are implemented as in (29), the characteristics of the optimization are not necessarily fixed. By tuning the weighting matrices different characteristics of the optimized thruster commands may be obtained, which will be shown in section IV and V. Before analysing how the tuning affects the performance of the proposed thrust allocation algorithm, the thrust allocation algorithm is to be compared to a more standard thrust allocation algorithm in a benchmarking test.

III. Benchmarking

In order to test the proposed thrust allocation algorithm a one-step algorithm, similar to the one presented in [16], is to be used for comparison. The reason why this algorithm is called a one-step algorithm is because the optimization horizon consists of only one point. Here, the horizon time is T=1 s, having only one sample. It might seem a bit unfair to compare a one-step optimization algorithm to an MPC algorithm, however, such one-step optimization algorithms are often used in industrial applications and, hence, suited to be used for comparison. It is expected that the proposed thrust allocation algorithm will outperform the one-step algorithm regarding reducing thrust rates and thruster angle rates, while reducing the total power consumption. However, it is also expected that the one-step algorithm will be faster than the proposed algorithm. Hence, the total energy consumption for the two algorithms, as well as the mean computational time, are to be compared as two of the key-parameters in this study.

The one-step optimization thrust allocation algorithm used for comparison is given as

$$\min_{\boldsymbol{F}_c, \boldsymbol{\alpha}_c \in \mathcal{R}} J(\boldsymbol{x}, \boldsymbol{x}_d, \boldsymbol{\Delta\alpha}, \boldsymbol{F}_c, \boldsymbol{\Delta F}_c) \tag{32a}$$

subject to

$$\underline{\boldsymbol{F}} \leq \boldsymbol{F}_c \leq \overline{\boldsymbol{F}} \tag{32b}$$

$$\underline{\boldsymbol{\Delta F}} \leq \boldsymbol{\Delta F}_c \leq \underline{\boldsymbol{\Delta F}} \tag{32c}$$

$$\underline{\boldsymbol{\Delta\alpha}} \leq \boldsymbol{\Delta\alpha}_c \leq \overline{\boldsymbol{\Delta\alpha}} \tag{32d}$$

where

$$\begin{aligned} J(\cdot) = (\boldsymbol{x} - \boldsymbol{x}_d)^\top \boldsymbol{Q}_e (\boldsymbol{x} - \boldsymbol{x}_d) + \boldsymbol{\Delta\alpha}^\top \boldsymbol{Q}_{\Delta\alpha} \boldsymbol{\Delta\alpha} \\ + \boldsymbol{F}_c^\top \boldsymbol{Q}_F \boldsymbol{F}_c + \boldsymbol{\Delta F}_c^\top \boldsymbol{Q}_{\Delta F} \boldsymbol{\Delta F}_c \end{aligned} \tag{33}$$

and $\boldsymbol{x}$ is as defined in (4), $\boldsymbol{x_d}$ is the corresponding reference given by the DP-controller, $\Delta\boldsymbol{\alpha}$ is a vector containing the thruster angle rates for the two thrusters placed at the stern of the vessel, and is calculated as the difference between the previous output of the algorithm and the current output of the algorithm. $\boldsymbol{F}_c$ is a vector containing the three thrust amplitudes for the thrusters, $\Delta\boldsymbol{F}_c$ is the thrust amplitude rates for the three thrusters, calculated the same way as the thruster angle rates, $\underline{\boldsymbol{F}_c}$, $\underline{\boldsymbol{\Delta F}_c}$ and $\underline{\boldsymbol{\Delta\alpha}_c}$ are the limit vectors for the vectors $\boldsymbol{F}_c$, $\boldsymbol{\Delta F}_c$ and $\boldsymbol{\Delta\alpha}_c$, respectively. $\boldsymbol{Q}_e$ is the cost matrix for the error in global thrust, $\boldsymbol{Q}_{\Delta\alpha}$ is the cost matrix for the thruster angle rates, $\boldsymbol{Q}_F$ is the cost matrix for the thrust amplitudes and $\boldsymbol{Q}_{\Delta F}$ is the cost matrix for the thrust amplitude rates.

To simulate the performance of the two thrust allocation algorithms, the vessel model derived in [26] is to be utilized. This vessel model is briefly presented in section III-A in order to demonstrate the model fidelity.

TABLE I
Main parameters in vessel model

Parameter	Description	Value
L	Length of ship	$107\,m$
B	Width of ship	$22\,m$
D	Draught of ship	$5\,m$
$P_{m,max}$	Power saturation main thrusters	$3.5\,MW$
$P_{b,max}$	Power saturation bow thruster	$3.5\,MW$
v_{cN}	Northward current	-0.1 m/s

A. Vessel Model

The vessel model contains a DP-controller, a wave filter [25], current and simple hydrodynamics based on wave potential theory, including second order mean drift forces and irregular sea states [30], as well as thruster models. The thruster configuration is the same as shown in Fig. 4, e.g. two main thrusters symmetrically placed at the stern and one tunnel thruster in the bow, and the produced thrust is assumed measurable. Note that the main thrusters are rotatable and the bow thruster is fixed and produces thrust only in the sway direction. The main vessel parameters and thruster parameters are given in TABLE I and II, respectively. Note that TABLE II sets the restrictions in (32b)-(32d).

TABLE II
Main parameters describing the thruster configuration as given in Fig. 4

Thruster	Parameter	Description	Value
Main thruster port side (PS) and starboard (SB)	i	Thruster ID	PS :1, SB: 2
	(x, y)	Thruster position $[m]$	$(-45, \mp 7)$
	$\underline{F_{ic}}$	Min. thrust force $[N]$	-1000000.0
	$\overline{F_{ic}}$	Max. thrust force $[N]$	1000000.0
	$\underline{u_{ic}}$	Min. thrust rate $[\frac{N}{s}]$	-1000.0
	$\overline{u_{ic}}$	Max. thrust rate $[\frac{N}{s}]$	1000.0
	$\underline{\dot{\alpha}_i}$	Min. angular rate $[\frac{^\circ}{s}]$	-10.0
	$\overline{\dot{\alpha}_i}$	Max. angular rate $[\frac{^\circ}{s}]$	10.0
	α_{ib}	Bias angle $[^\circ]$	0.0
Bow thruster	i	Thruster ID	3
	(x, y)	Thruster position $[m]$	$(53, 0)$
	$\underline{F_{ic}}$	Min. thrust force $[N]$	-1000000.0
	$\overline{F_{ic}}$	Max. thrust force $[N]$	1000000.0
	$\underline{u_{ic}}$	Min. thrust rate $[\frac{N}{s}]$	-1000.0
	$\overline{u_{ic}}$	Max. thrust rate $[\frac{N}{s}]$	1000.0

The reader is referred to [26] for details regarding the vessel model. A total overview of the simulation model is given in Fig. 5. In the figure, $\boldsymbol{y}_m$ refers to the measurement vector including measurements of the vessel's position and heading, $\boldsymbol{y}_m^f$ is the filtered measurement vector, $\dot{\boldsymbol{y}}_m^f$ is the rate of the filtered measurement vector, $\boldsymbol{y}_d$ is the vector of the desired position and heading, $\dot{\boldsymbol{y}}_d$ is the corresponding rate vector, and τ_i is the thrust output from thruster i. Note that the thrust contributions from each thruster is transformed into global thrust contributions,

$$\boldsymbol{\tau}_g = \boldsymbol{H}(\alpha_1, \alpha_2)\boldsymbol{\tau} \tag{34}$$

where $\boldsymbol{\tau}_g \in \mathcal{R}^6$ is the global thrust vector, $\boldsymbol{H}(\cdot) \in \mathcal{R}^{6\times 3}$ is the thrust allocation matrix and $\boldsymbol{\tau} = [\tau_1,\ \tau_2,\ \tau_3]^\top$.

In order to compare the two different thrust allocation algorithms a manoeuvring test is designed and is elaborated in

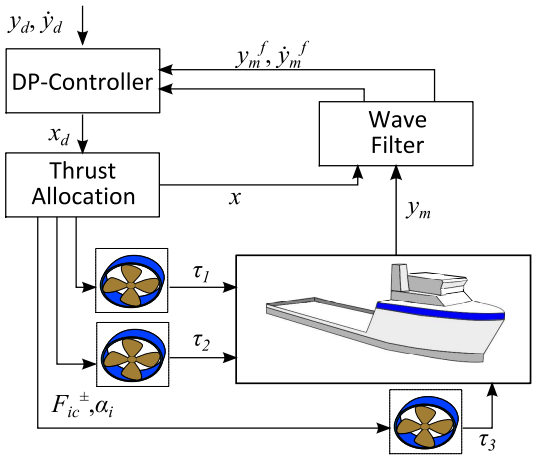

Fig. 5. Simulation setup

TABLE III
COST FUNCTION WEIGHTS IN THE PROPOSED THRUST ALLOCATION ALGORITHM AND THE ONE-STEP TRUST ALLOCATION ALGORITHM. (i,i) DENOTES THE ENTIRE DIAGONAL OF THE MATRIX

MPC		**One-Step**	
$\boldsymbol{Q}_k(i,i)$	10.0	$\boldsymbol{Q}_c(i,i)$	100.0
$\boldsymbol{Q}_T(i,i)$	100.0	-	-
$\boldsymbol{Q}_{uF}(i,i)$	1500.0	$\boldsymbol{Q}_{\Delta\alpha}(i,i)$	50000000.0
$\boldsymbol{Q}_F(i,i)$	1.0	$\boldsymbol{Q}_F(i,i)$	0.01
$\boldsymbol{Q}_u(i,i)$	500.0	$\boldsymbol{Q}_{\Delta F}(i,i)$	1.0

the following, along with the choice of cost function weights in the two thrust optimization formulations.

B. Simulation Setup and Tuning

Both the MPC thrust allocation algorithm and the one-step thrust allocation algorithm are tuned to perform as good as possible and to minimize both energy consumption and large oscillations in the commands due to environmental disturbances such as waves, while maintaining stability and robustness. TABLE III shows a summary of all the cost function weights. Note that the MPC thrust allocation algorithm is tuned a bit harder, having in general higher costs for the produced thrust and the thrust rates, because it is more robust due to the optimization horizon. Hence, it is expected that the proposed thrust allocation algorithm would have an additional advantages in this benchmarking test.

The environmental forces acting on the vessel in this simulation are the northward current and irregular waves, and the main parameters describing these environmental forces are given in TABLE IV. In the simulation, the vessel is heading north, initially, and is to move $20\,m$ to the north before changing the heading so that it faces east. Then, the vessel is to move $20\,m$ to the east, before changing heading facing south and moving $20\,m$ to the south. Afterwards, the heading is changed to west before the vessel moves $10\,m$ to the west. Lastly, the heading is changed such that the vessel faces northwards before finally moving $10\,m$ to the north. Note that filters are used to smooth the reference signals before being fed to the DP-controller.

TABLE IV
ENVIRONMENTAL FORCES FROM IRREGULAR SEA STATE

Parameter	**Description**	**Value**
H_s	Significant wave height	$1.0\,m$
T_p	Wave peak period	$8\,s$
N_w	Number of wave components	50 -
γ	Jonswap-spectrum parameter	3 -
T_d	Lower wave spectra period	$0.2\,s$
T_u	Upper wave spectra period	$50\,s$

For the proposed algorithm, the integrator time step is set as $\Delta t = 0.005\,s$, the optimization horizon time step is set as $\Delta t_k = 1.0\,s$, and the horizon is set to $T = 30\,s$, resulting in 30 samples in the horizon. The reason for setting the horizon to $30\,s$ is because of the rate limitations for the thruster angles. The thrusters should at least be able to rotate $180\,^\circ$ during the horizon in order to have the possibility to either reverse the thrust or to rotate the thrusters $180\,^\circ$. Hence, with a maximal angular rate of $\pm 10.0\,^\circ\!/s$ the horizon could have been set to T=18.0 s, but because of robustness reasons the horizon is set larger. However, this will be studied in more detail in section IV.

The optimization parts in the thrust allocation algorithms are run every Δt_k seconds and only the results from the first sample in the MPC algorithm, $k = 1$, in the horizon are used. The simulation time is set to $4500\,s$, and the DP-control system is initiated at t =$30\,s$. The simulation results are shown in the following subsection.

C. Simulation Results

Fig. 6 shows the vessel position and orientation for the two cases. Note that the abbreviation OS is used for the one-step thrust allocation algorithm. As can be seen in the figure, the simulation results show that the vessel position and orientation from the two cases converge, and it is not possible to distinguish the cases from each other. This indicates that both algorithms are equally good at keeping the vessel in position. This can be verified in Fig. 7 which shows the error between the commanded and the measured positions and orientations. Also in this figure the simulation results seem to converge, except for some small differences. The simulation results also indicate the performance of the DP-controller, which seems to be stable and able to add an appropriate amount of damping to the vessel motion.

Closely related to the outputted thruster commands from the thrust allocation algorithms are the power consumptions from the thruster systems which are shown in Fig. 8. In the figure, the upper two plots show the power consumption for the two main thrusters, while the third plot shows the power consumption for the bow thruster. The last plot shows a comparison of the total power consumption of the entire propulsion system for the two thrust allocation algorithms.

In the beginning and the end of the simulation, where the vessel is to keep a stationary position and orientation, the power consumptions seem to overlap, having a total power consumption of about $16\,kW$. However, as the figure shows, the power consumptions increase significantly when the vessel is facing east or west. This has to do with the orientation of

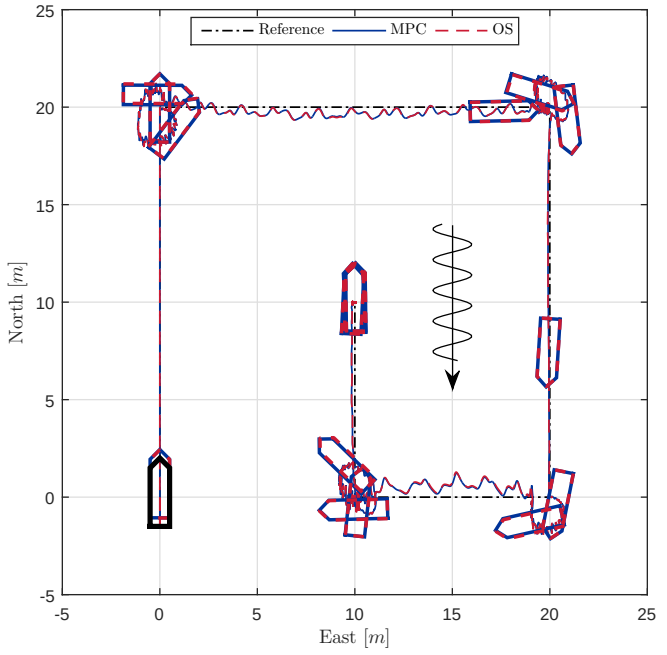

Fig. 6. North-East plot including heading. The thick black graph in the lower leftmost corner in the figure denotes the initial position and orientation of the vessel

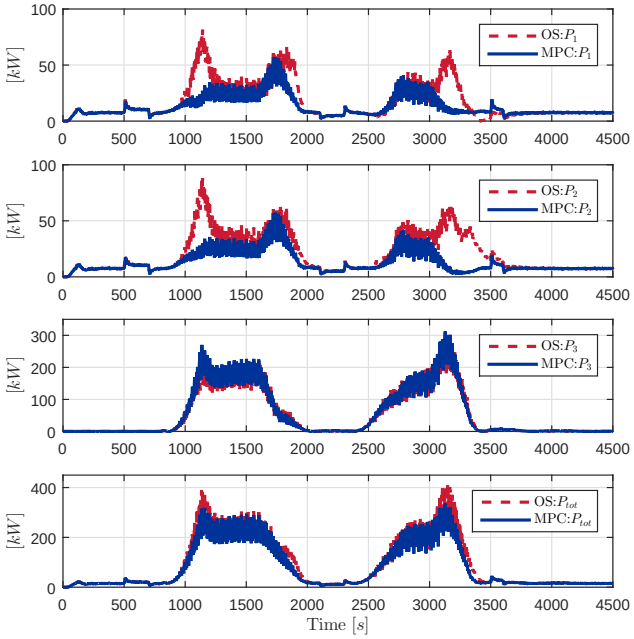

Fig. 8. Comparison of thruster power and total power consumption of thruster system for the two thrust allocation algorithms

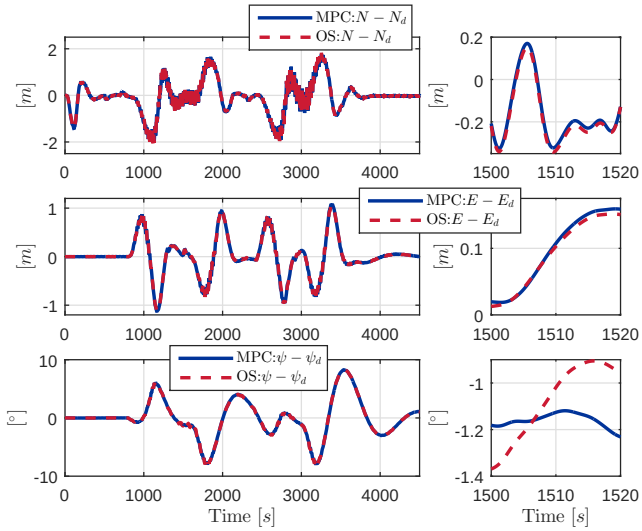

Fig. 7. Comparison of position and orientation errors for the two thrust allocation algorithms. MPC denotes the MPC-based thrust allocation algorithm while OS denotes the One-Step thrust allocation algorithm

the ship in comparison to the angle of attack for the waves and the current forces. Hence, when the vessel is positioned east-westwards, the environmental forces acting on the ship grows significantly since the forces attack the heel of the vessel. From the figure it is also possible to see that the power consumption in this case for the two main thrusters are higher for the one-step algorithm than for the proposed algorithm. For main thruster 2, the maximal power consumption is 88 kW for the one-step algorithm, while only 57 kW for the MPC algorithm. In total, the maximal power consumption for the one-step algorithm is about 414 kW, while 336 kW for the MPC algorithm. Hence, it is not surprising that the one-step algorithm has a higher energy consumption than the MPC algorithm, about 114.6 kWh in comparison to 104.3 kWh for the MPC algorithm. This means that the one-step algorithm consumes about 10 % more energy than the MPC algorithm in this simulation case. Also, it seems like the oscillations in the power consumption is slightly reduced in the MPC algorithm in comparison to the one-step algorithm, which also can be verified in Fig. 9.

Fig. 9 shows the comparison between the commanded thrust references and the commanded thruster angles for the two main thrusters. As can be seen in the figure, the two algorithms output about the same thrust- and angle commands between the start of the simulations and to about 800 s. After 800 s the vessel starts moving eastwards and both algorithms rotate both the main thrusters counter clockwise, while the produced thrusts are increased significantly when the vessel is oriented in an east-westward direction. This is repeated during the entire manoeuvre. At 3000 s the one-step thrust allocation algorithm decides to rotate main thruster 1 additionally 180 ° and to reverse the corresponding thrust in comparison to main thruster 2 and the main thrusters in the MPC thrust allocation algorithm. However, this is not done in one optimization step, but over a time span of 300 s, which indicates that the resulting commands from the one-step algorithm are not affected by numerical errors due to a low number of allowed iterations, or poor KKT-conditions. On the contrary, it is believed that there exist multiple local optimal minima because of the non-linearities in the one-step problem formulation, and thus, a solving procedure including multiple shooting strategies should be considered if such an algorithm is to be implemented

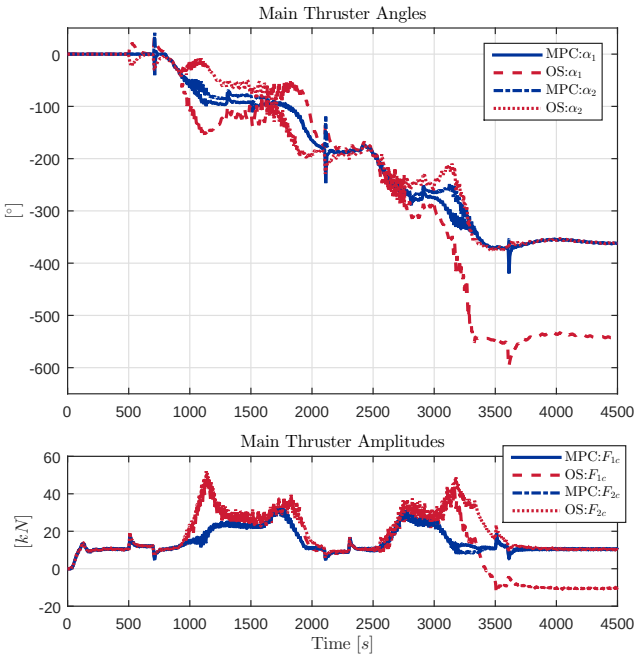

Fig. 9. Comparison of thruster orientations and thrust amplitudes for the two main thrusters

in a realistic manner. However, these results do not affect the total power consumption.

In the simulation time span t =3500 s to the end of the simulation, the two thrust allocation algorithms output about the same commands when taking into consideration that the one-step algorithm has reversed main thruster 1 and that the corresponding thrust command is mirrored. Another interesting observation is that the thrusters in the MPC-case rotates in total 360 ° during the simulation, and thus follows the rotation of the vessel. This illustrates the smoothness of operation of the MPC algorithm. It can also be seen in the figure that both the thruster angle commands and the thrust commands are oscillating less in the MPC algorithm in comparison to the one-step algorithm.

In summary, this benchmarking test has shown that the proposed algorithm has the potential to outperform the one-step algorithm, both when it comes to reduced energy consumption and reduced oscillations in the thrust- and thruster angle commands. When it comes to computational speed, it is of no surprise that the one-step algorithm is faster than the MPC algorithm. The mean computational time for the one-step algorithm in this benchmarking test is 0.385 ms while the mean computational time for the MPC algorithm is 10.56 ms. This means that the one-step algorithm is about 27.4 times faster than the MPC algorithm in this case. However, when having in mind that the horizon of the MPC algorithm is 30 times larger than in the one-step algorithm, the resulting mean computational speed for the MPC algorithm is fast, and since the optimization in the MPC algorithm is initiated only every second, the total algorithm is about 94.7 times faster than real-time. The main results are summarized in TABLE V.

TABLE V
SIMULATION RESULTS FROM BENCHMARKING TEST

KPI	OS	MPC
Maximal power consumption	414 kW	336 kW
Energy consumption	114.6 kWh	104.3 kWh
Mean computational time	0.385 ms	10.56 ms

In the last part of this article the MPC algorithm is to be studied further with respect to built-in filtering properties. Before testing the algorithm with different cost function weights in a realistic simulation case, some preliminary tests are performed in order to map the different properties regarding tuning of cost function weights and the length of optimization horizon.

IV. Preliminary Case Studies

In the preliminary case studies, a response test is to be applied to the thrust allocation algorithm. The reference signal, $\boldsymbol{x}_d$, from a potential DP-controller, only contains a surge thrust reference, meaning that $Y_d = Mz_d = 0.0$. The surge thrust reference that is fed to the thrust allocation algorithm consists of a ramp-up and a ramp-down, as well as a step-up and a step-down. Also, the simulation settings and the thrust allocation algorithm time steps are set as in the benchmarking test. The same thruster configuration as used in the benchmarking test, as shown in Fig. 4, is to be used. The main parameters describing the thruster configuration are the same as listed in TABLE II, except that now the thruster biasing angle is set to $\pm 20\,^\circ$.

It is expected that the tuning of the cost function weights would have a significant impact on the performance of the proposed algorithm. Hence, as a result of proper tuning, it is expected that the proposed algorithm can be tuned such that thrust rates and thruster angle rates are reduced in order to decrease wearing of the propulsion system. When it comes to optimization horizon lengths, the length should be at least long enough for the algorithm to be able to consider whether it is optimal to rotate a thruster or to reverse the corresponding thrust. In this case it means that the optimization horizon should be at least T = 18 s because of the angle rate limitations. Hence, it is expected that the main advantage by increasing the horizon even further is gained robustness.

The first preliminary case study treats the cost function weights and is elaborated in the following.

A. Cost Function Weights

Three different sets of weighting matrices are to be tested and compared for an optimization horizon of 30 s. Only changes in four of the weighting matrices are considered, namely $\boldsymbol{Q}_k$, $\boldsymbol{Q}_{uF}$, $\boldsymbol{Q}_F$ and $\boldsymbol{Q}_u$. The weights for the three tuning cases are set as in TABLE VI. Note that case 3 has the same weights as used in the benchmarking test and that all entries in the weighting matrices that are not given in the table are set to zero. The results from the three different cases are compared in Fig. 10 and 11, showing the thrust allocation algorithm output commands and the first horizons in the simulations, respectively.

TABLE VI
WEIGHTING MATRICES IN DIFFERENT CASES. (i,i) DENOTES THE ENTIRE DIAGONAL OF THE MATRIX

Weight	Case 1	Case 2	Case 3
$\boldsymbol{Q}_k(i,i)$	100.0	100.0	10.0
$\boldsymbol{Q}_T(i,i)$	100.0	100.0	100.0
$\boldsymbol{Q}_{uF}(2,2)$	10.0	500.0	1500.0
$\boldsymbol{Q}_{uF}(4,4)$	10.0	500.0	1500.0
$\boldsymbol{Q}_F(i,i)$	0.1	0.1	1.0
$\boldsymbol{Q}_u(i,i)$	10.0	20.0	500.0

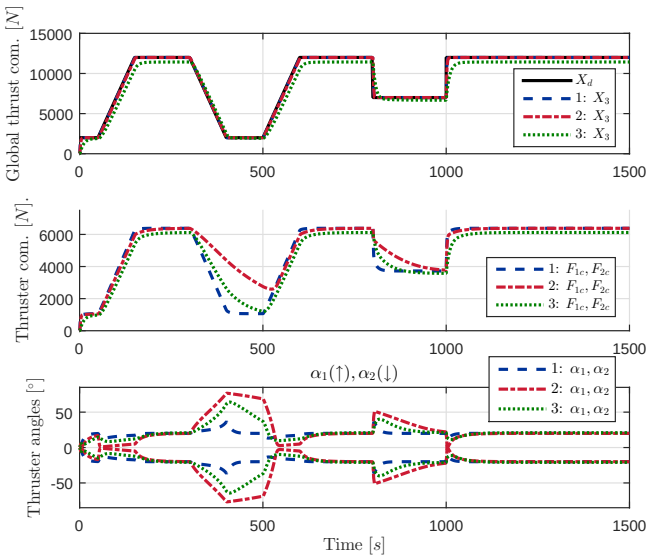

Fig. 10. Global thrust forces, thruster commands and thruster angle commands for the three different cases. Note that α_1's are plotted in the upper part of the plot (↑), and that α_2's are plotted below (↓)

The first plot in Fig. 10 shows that the three different cases more or less overlap the reference, X_d, except for case 3 which is slightly lower. This has to do with the reduction in the cost function weights $\boldsymbol{Q}_k$ in comparison to the two other cases. However, since a DP-control law with integral effect is used, this will not result in bad performance, which has already been established in the benchmarking test. The second plot in Fig. 10 shows the characteristics of the optimized thrust commands, F_{1c} and F_{2c}, for the three different cases. As expected, the first case produces the fastest thrust commands, while the second case produces the slowest. The third plot in the figure shows that when the thrust rates decrease, the thruster angles must compensate for that. It is clear from the plot that case two allows more thruster biasing than the two other cases.

The results for the port-side main thruster from the first solved horizon in each case are compared in Fig. 11. As can be seen in the figure, all the thrust magnitudes from the three cases seem to converge during the time horizon. However, some significant differences between the cases can be seen. The first case seems to focus more on reaching a desired thrust before starting to increase the angle in comparison to the two other cases. It can also be seen that the thrust rate in the first case is much higher than in the two other cases, causing the thrust magnitude to reach its desired value faster

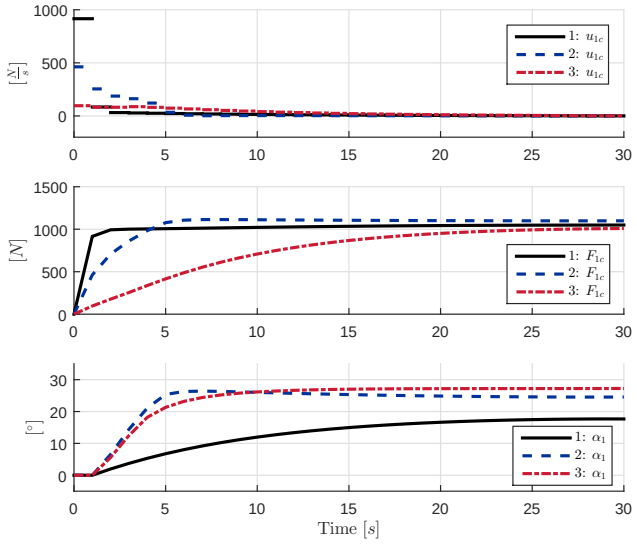

Fig. 11. Comparison of the first solved time horizon for port-side main thruster for the three cases

than in the two other cases. When it comes to computational speeds, case 1 is fastest having a mean computational speed of 5.68 ms for each optimization step. Secondly, case 3 had a mean computational speed of 9.08 ms for each optimization step, while case 2 was the slowest one with a speed of 10.4 ms.

The results from this preliminary case study show that by increasing the cost function weights $\boldsymbol{Q}_{uF}$ and $\boldsymbol{Q}_u$ thruster biasing is becoming optimal and reduces the maximal thrust rates. This would be an important property when considering reducing wear of the total propulsion system and the power plant, as well as the amount of consumed energy, which is studied in section V. In the next preliminary case study two different optimization horizon lengths are compared.

B. Length of Horizon

In section III-B it was stated that the optimization horizon should be at least 18 s in order to allow the algorithm to determine whether to rotate a thruster or to reverse the corresponding thrust if the thruster could rotate with a maximal angular rate of 10 °/s. However, for robustness reasons, the optimization horizon was set to 30 s. In this preliminary case study, both these two lengths of optimization horizons are tested and compared with the cost function weights for case 1 in TABLE VI. The results from the two different optimization horizons are compared in Fig. 12 and 13, showing the thrust allocation algorithm output commands and the first horizons in the simulations, respectively.

As can be seen in Fig. 12, both global thrust commands overlaps the thrust reference. Also, in the second plot it is hard to distinguish the thruster commands from the two different optimization horizon lengths. However, some difference can be seen in the thruster angles. The thruster angles in the case with a horizon of 30 s seem to be slightly larger in magnitude in the peaks. Nevertheless, from these results it can be concluded that the two different horizons perform equally in this case.

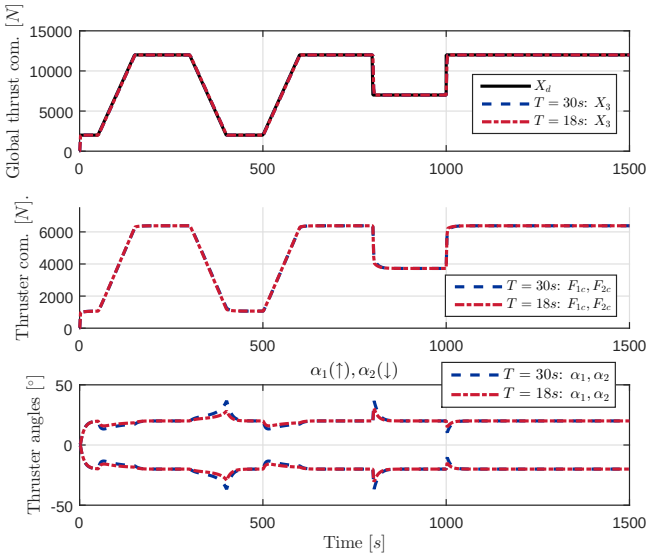

Fig. 12. Global thrust forces, thruster commands and thruster angle commands for the two different optimization horizon sizes. Note that α_1's are plotted in the upper part of the plot (↑), and that α_2's are plotted below (↓)

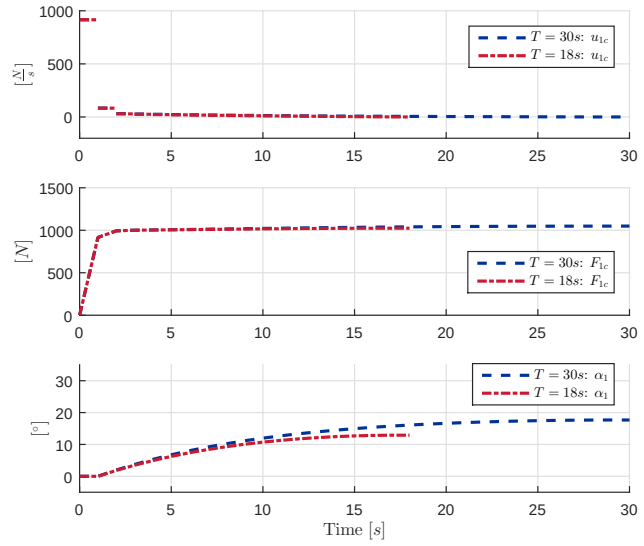

Fig. 13. Comparison of the first solved time horizon for port-side main thruster for the two different optimization horizon sizes

The results for the port-side main thruster from the first solved horizon in each case are compared in Fig. 13. As can be seen in the figure, the results show that the two different optimization horizons overlap in the beginning of the simulation, but the case with the shortest horizon converges faster due to the shorter horizon. Also, the thruster angles seem to be slightly larger in the case with the longest horizon, as was also the case in Fig. 12. When it comes to computational speed, the thrust allocation algorithm with the shortest horizon had a mean computational speed of 1.65 ms for each optimization step, while the other case had a computational speed of 5.68 ms.

In summary, the results show that an optimization horizon of 18 s seems to perform equally good as an optimization horizon of 30 s, mostly due to the maximal allowed angular rates, but also being about 3.4 times faster. However, since the thrust allocation algorithm with an optimization horizon of 30 s is much faster than real-time, this horizon is considered in the rest of the work due to robustness reasons, giving the algorithm even more time to consider rotating or reversing a thruster.

Based on the results obtained in these preliminary case studies, a case study showing the effect of using different cost function weights with respect to reduced oscillations in thruster commands and power consumption in a vessel manoeuvring operation is to be performed.

V. Main Case Study

The results from the benchmarking case in section III and the preliminary case studies in section IV indicate that even though the implementation of the proposed thrust allocation algorithm is fixed, the characteristics of the algorithm can be tailored by tuning the cost function weights. When keeping a vessel in dynamic positioning operations, the wave filter plays an important role in reducing the power consumption, as well as wearing of the power systems, during the operation. This, through filtering out the small wave contributions that keeps the vessel oscillating around its set-position and to catch the drift effects and the slowly varying forces.

Although a wave filter is included and tuned as best, it is impossible to filter out all small oscillatory wave contributions. These contributions are given as input to the DP-controller which tries to compensate for them. This was also the case in the benchmarking test, which showed that even though a wave filter was used, there still was oscillations in the thruster commands, and, hence, the power consumption. However, it is believed that by tuning the thrust allocation algorithm properly, one could obtain an optimal solution with respect to the goal at hand, namely that the thruster angles compensate for these contributions instead of varying the thrust amplitudes. This will be the topic in this case study.

A. Simulation Setup

The generic offshore vessel model presented in [26] is to be used for testing the three different tuning cases of the thrust allocation algorithm also in this case. The main parameters describing the vessel model and the environmental forces are the same as in the benchmarking test and are given in TABLE II and IV, respectively.

In these simulations, the vessel is to follow a zig-zag trajectory, as shown in Fig. 14. Initially, the vessel is to keep its position at $(0, 0)$ in North-East coordinates with the bow facing north, before slowly moving 20 m to the north. Following, the vessel changes heading to face east before moving 20 m eastwards. Afterwards, the vessel changes the heading back to due north before moving additionally 20 m northwards. When the new position has been reached, the vessel changes heading to face west, before moving 40 m westwards. Again, the vessel changes the heading back to due north before moving additionally 20 m to the north. Following, the heading

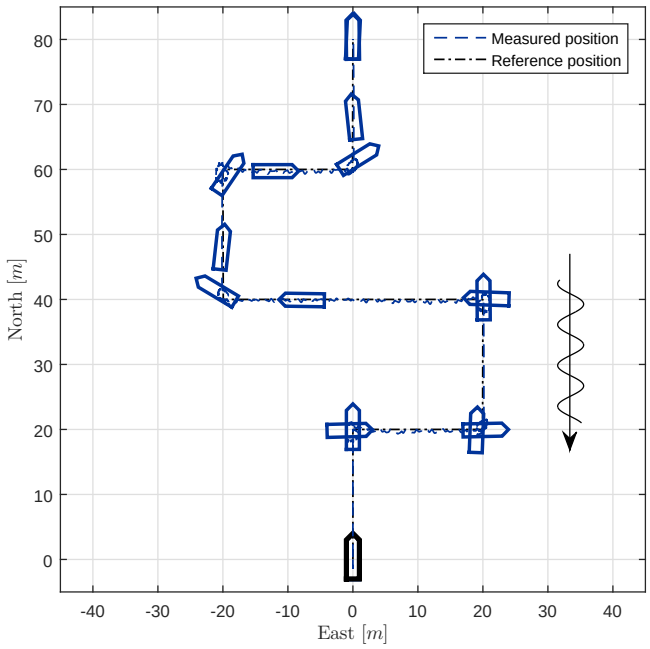

Fig. 14. North-East plot including heading. The thick black vessel outline denotes the initial position and orientation of the vessel

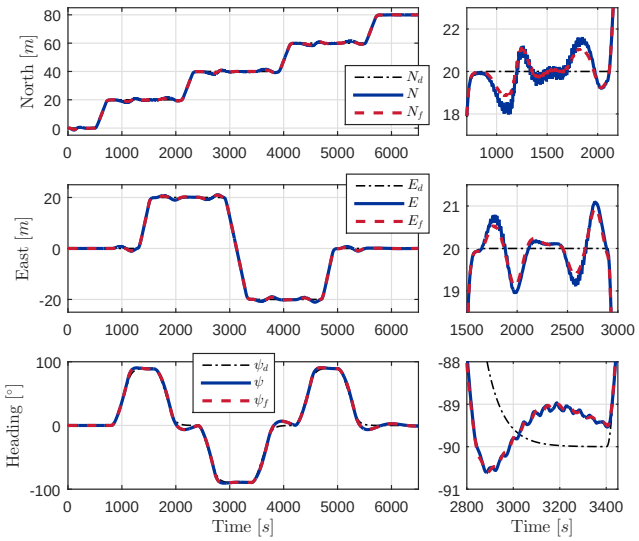

Fig. 15. Position and orientation of the vessel, both references, measurements and filtered measurements. Note that the small plots to the right in the figure shows a magnified region of the plots to the left

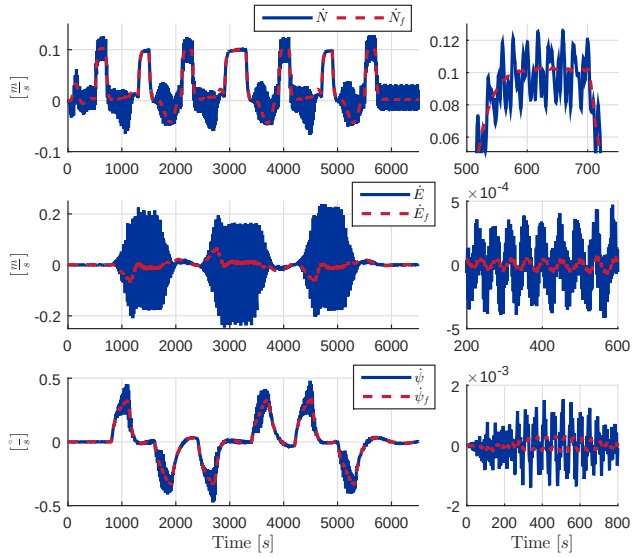

Fig. 16. Position and orientation rates of the vessel, both measurements and filtered measurements. Note that the small plots to the right in the figure shows a magnified region of the plots to the left

of the vessel is changed to again face east before moving 20 m eastwards. Finally, the vessel changes heading back to due north before moving to the final position 20 m northwards. It is expected that this manoeuvring will stress test the algorithm and the wave filter such that different effects can be reflected in the simulation results as well as ensuring robustness for the proposed algorithm. The simulation results are presented in the following.

B. Simulation Results

Before presenting the diverging simulation results from the three different tuning cases, the converging simulation results are given. These results include the position and the orientation of the vessel, the global thrust commands compared to the global optimal thrust commands from the thrust allocation algorithm, and the filtered measurements from the wave filter. For the sake of order, the data in the following figures have been obtained from the first tuning case.

Fig. 14 presents the position and the heading of the vessel in a north-east plot including the heading of the ship for given time steps. As can be seen in the figure, the vessel seems to keep its position and heading quite well in addition to follow the new position commands. These observations are verified by Fig. 15 which compares the measurements ($_m$) to the commands ($_d$) and the filtered measurements ($_f$). The results in this figure also indicate that the wave filter works well in filtering out most of the fastest oscillatory effects from the waves, as can be seen in the magnified plots to the right in the figure. The same can be concluded when looking at the filtered position rates and the heading rate compared to the corresponding measurements in Fig. 16, although there still are some oscillations present in the filtered states. These oscillations will be fed to the DP-controller and cause additionally oscillations in the power consumption if not being suppressed by the thrust allocation algorithm.

In Fig. 17 the output from the DP-controller is compared with the corresponding output from the thrust allocation algorithm. As can be seen in the figure, the global thrust signals, $\boldsymbol{x}$, converge to the commanded global thrust signals, $\boldsymbol{x}_d$, except for when the commanded rate magnitudes become too high as can be seen in the comparison between Y_d and Y_3. However, the thrust allocation algorithm seems to handle such limitations as it is supposed to.

Even though the commanded global thrust from the DP-

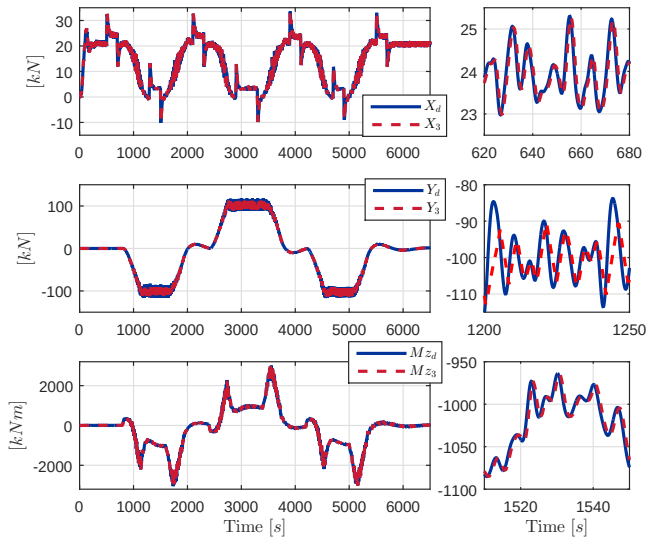

Fig. 17. DP-commands compared to global optimal thrust from the thrust allocation algorithm. Note that the small plots to the right in the figure shows a magnified region of the plots to the left

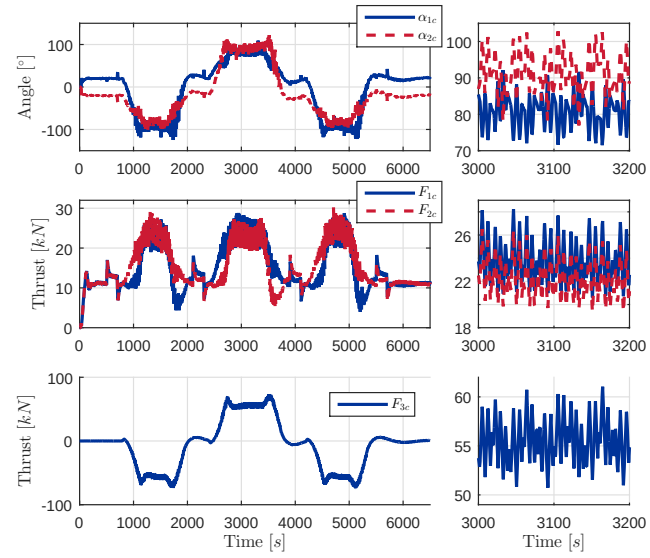

Fig. 18. Thrust commands and angle commands compared to measured thrusts and angles for case 1. Note that the small plots to the right in the figure shows a magnified region of the plots to the left

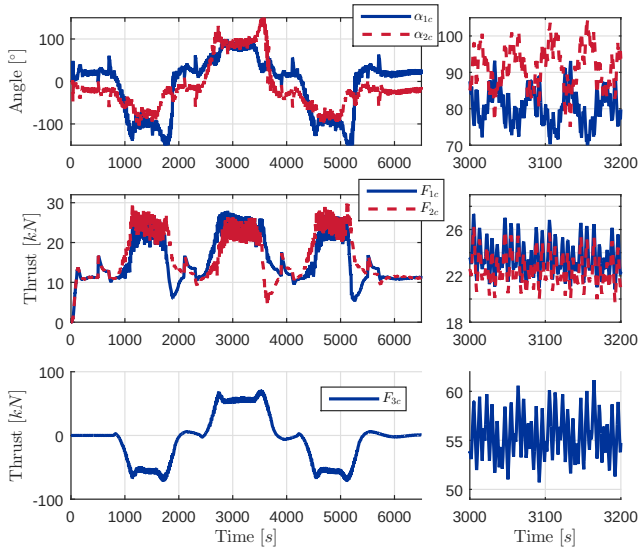

Fig. 19. Thrust commands and angle commands compared to measured thrusts and angles for case 2. Note that the small plots to the right in the figure shows a magnified region of the plots to the left

controller and the corresponding global thrust from the thrust allocation algorithm overlap in the three cases, the optimal thruster commands and the thruster angle commands are different. Fig. 18, 19 and 20 show these results for case 1, case 2 and case 3, as given in TABLE VI, respectively.

For case 1, Fig. 18 shows that the thruster angles have fast oscillations with small maximal amplitudes of about 5 °, in addition to slower oscillations with an amplitude of about 10 °, when the waves and current encounter the vessel's heel. Also, the main thrusters oscillate with a thrust amplitude of about 3.5 kN as well when the vessel is oriented east-westwards. The thruster angles seem to follow their biasing angles quite good except for when a vessel position change is initiated or when the vessel is parallel with the wave beam. Since the bow thruster is fixed, the thrust command signal seems to oscillate quite a bit as well having an amplitude of about 5 kN. As will be seen later on, the results from the bow thruster can also be improved by tuning the thrust allocation algorithm properly, even though there is no controllable thruster angle to work with.

Fig. 19 shows the corresponding results from case 2. As can be seen in the figure, the fastest oscillations in the thruster references are decreased to about 2.5 kN in magnitude, a reduction of about 30 % compared to the results from case 1. However, the amplitude of the oscillations in the thruster angles are quite different in this case in comparison to case 1. In the beginning of the simulation the amplitude is increased from about 1 ° to about 7.5 ° when compared to the first case. When the vessel is oriented east-westwards the picture is different. The amplitude of the fastest oscillations in the thruster angles are decreased to about 4 ° while the amplitude of the slowest oscillations are increased to about 15 °. It is also possible to see that thruster biasing is more accepted in this cases when looking at the peak in the thruster angle command α_2 around t = 3500 s. The figure also shows that the fastest oscillations in the thruster commands for the bow thruster are reduced to about 3.5 kN.

The results from case 3 are given in Fig. 20. The results show that the oscillations in the thrust are decreased even more, to about 1 kN in amplitude, a reduction of about 70 % in comparison to the results from case 1. Also the oscillations in the thrust angles are decreased in the entire simulation, to about 1 ° in the beginning of the simulation as well as for the fastest oscillations when the vessel is oriented east-westwards, while the maximal amplitude of the slowest oscillations is

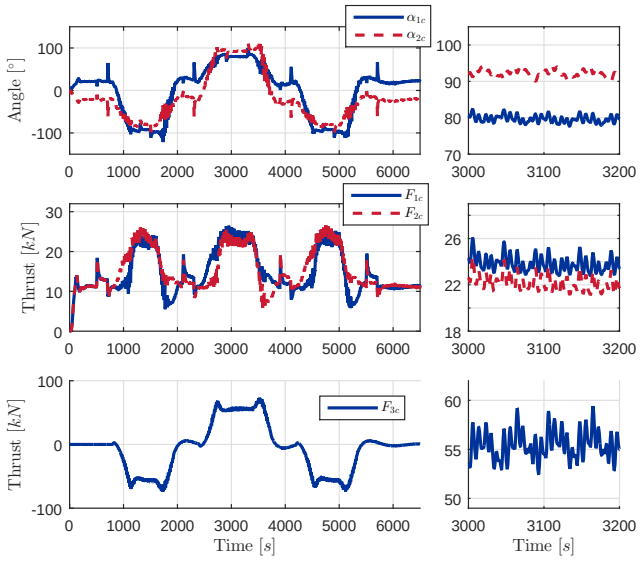

Fig. 20. Thrust commands and angle commands compared to measured thrusts and angles for case 3. Note that the small plots to the right in the figure shows a magnified region of the plots to the left

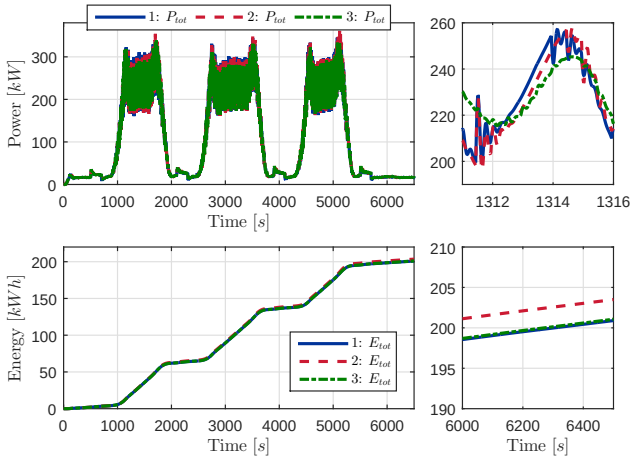

Fig. 21. Comparison of power consumption and total energy consumption for the three simulation cases. Note that the small plots to the right in the figure shows a magnified region of the plots to the left

decreased to about 2 °. The reason for this is that the weights in $\boldsymbol{Q}_{uF}$ and $\boldsymbol{Q}_u$ are increased significantly in comparison to the error weights in $\boldsymbol{Q}_k$ and $\boldsymbol{Q}_T$. This means that the thrust allocation algorithm allows more error than in the two previous cases. However, the errors that are allowed are only due to the small oscillations, meaning that the chosen weight combination in the thrust allocation algorithm works more or less as a low pass filter without phase differences. This error is completely fine since the integrator in the DP-controller makes sure that the vessel doesn't experience a drift-off due to this error.

When it comes to the fastest oscillations in the bow thruster commands, the maximal amplitude is reduced even more to about 2 kN, which lead to a smoother power consumption, as can be seen in Fig. 21. In case 1 the total power consumption

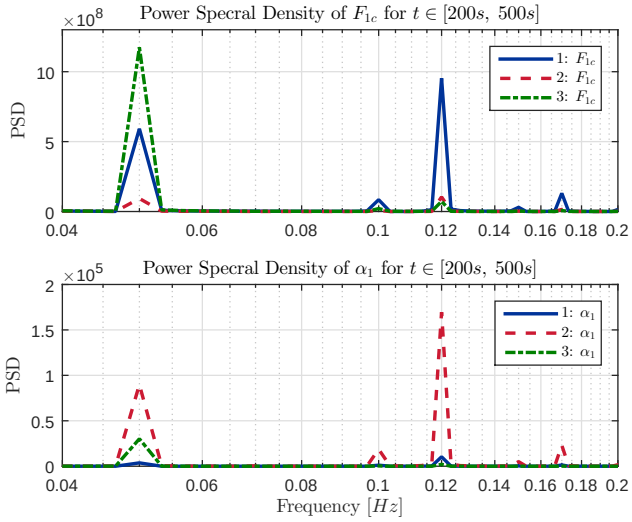

Fig. 22. Power spectral density of commanded thrust force and orientation for main thruster 1

was about 200.9 kWh, and the increase of consumed power in case 2 and 3 were 1.3 % and 0.1 %, respectively. Also, it can be seen that the noise in consumed power is significantly reduced in the third case in comparison to the two others. This means that the extra fuel cost for tuning the thrust allocation algorithm to filter out environmental disturbances that have not been suppressed by the wave filter is negligible.

The results regarding the filtering properties for the thrust allocation algorithm are summarized in Fig. 22, showing a comparison of the power spectral density of the commanded thrust force and orientation of thruster 1. As the figure indicates, the fastest thrust oscillations are largest in simulation case 1 while the fastest thruster angle oscillations are largest in simulation case 2. However, the slowest oscillations in the thrust commands are largest in case 3 while the slowest thruster angle oscillations are largest in case 2. This means that case 1 and case 3 is better at compensating for the slowly varying environmental disturbances while case 2 filter out as much as possible by using the thruster angles. In summary these results show that the set of cost function weights used in case 3 would be the most preferred ones since it reduces the fastest oscillations in the thruster commands as well as in the power consumption, while being able to compensate for slowly varying environmental disturbances. This, in addition to keeping the total energy consumption at a minimum. The simulation results from this case study, as well as from the benchmarking test and the preliminary case studies, also argue for the use of thruster biasing in order to reduce wearing of the systems, even though it increases the total energy consumption.

The mean computational speeds in these simulations are given as 9.06 ms, 16.37 ms and 15.60 ms for case 1, 2 and 3, respectively. Even though case 2 is the slowest one, it is still about 61 times faster than real-time.

VI. Conclusion

In this work a proposed non-angular MPC-based thrust allocation algorithm has been presented and tested. The reason for formulating the optimization problem in thrust vectors instead of thrust amplitudes and angles, was to reduce non-linearities in the thrust allocation problem in order to make the thrust allocation algorithm fast solvable, at least in real-time, which was accomplished.

The proposed thrust allocation algorithm was benchmarked against a commonly used one-step thrust allocation algorithm, and the results showed that the proposed thrust allocation algorithm outperforms the one-step algorithm, both when it comes to total energy consumption as well as reducing oscillations in the thrust commands and thruster angle commands. Even though the one-step algorithm was 24.7 times faster than the MPC algorithm, the MPC algorithm had a real-time index (RTI) of about 94.7 in the benchmarking case. The benchmarking test also showed that the proposed thrust allocation algorithm worked smoothly when the commanded thruster angles increased, which indicates that the calculation procedures in Algorithm 1 and 2 work properly.

Three sets of rather coarse cost function weights were chosen in order to test the thrust allocation algorithm and to show different optimization strategies in the first preliminary case study. The sets of weights were then tested on a vessel model in order to see if the differences in weights affected the overall goal of keeping the vessel in position while reducing thrust oscillations, due to fast wave effects that could not be filtered out with the wave filter, by using thruster biasing and actively compensate for these fast oscillations with the thruster angles. The results showed that even though the weighting matrices were different, the position and orientation of the vessel were maintained. Also, the tuning of the proposed algorithm affected the total energy consumption of the propulsion system. It would be interesting in further work to include a power plant in the simulation, as the one presented in [31] which includes specific fuel consumption data, in order to include the power plant efficiency and to see how the tuning of the thrust allocation algorithm affects the total fuel consumption. Also, it should be mentioned that in the case study there are situations where the thruster angles approach orientations of $\pm 90\,^{\circ}$. In reality, such orientations would cause hydrodynamic thruster-thruster interactions, decreasing the efficiency of the main thruster affected by the wake of the other main thruster. However, such effects are not included in the vessel model, nor have restrictions for this been included in the thrust allocation algorithm, although it has been mentioned. This is left out for future works.

Even though the energy consumption increased negligibly from case 1 to case 3, the results also showed that thrust oscillations and thrust rates were decreased significantly from case 1 to 3. This points to the fact that the thrust allocation algorithm can be tuned such that wearing of the propulsion system and power plant can be reduced due to unfiltered environmental effects in the DP-controller commands.

Acknowledgment

The authors would like to thank the partners in the ViProMa project, Norwegian Research Counsel project number 225322, for providing financial support for studying distributed systems for marine systems and operations. The authors would also like to thank Professor Tor Arne Johansen for useful inputs on the presented work.

References

[1] T. A. Johansen and T. I. Fossen, "Control allocation - A survey," *Automatica*, vol. 49, no. 5, pp. 1087–1103, 2013.

[2] M. V. Kothare, V. Balakrishnan, and M. Morari, "Robust constrained model predictive control using linear matrix inequalities," *Automatica*, vol. 32, no. 10, pp. 1361–1379, oct 1996.

[3] C. E. Garcia, D. M. Prett, and M. Morari, "Model predictive control: Theory and practice-A survey," *Automatica*, vol. 25, no. 3, pp. 335–348, may 1989.

[4] D. Q. Mayne, "Model predictive control: Recent developments and future promise," pp. 2967–2986, 2014.

[5] J. M. Maciejowski, *Predictive control : with constraints*. Prentice Hall; 1 edition (September 6, 2000), 2002.

[6] L. Grüne and J. Pannek, *Nonlinear Model Predictive Control*, ser. Communications and Control Engineering. London: Springer London, 2011.

[7] A. Veksler, T. A. Johansen, R. Skjetne, and E. Mathiesen, "Thrust Allocation With Dynamic Power Consumption Modulation for Diesel-Electric Ships," *IEEE Transactions on Control Systems Technology*, vol. 24, no. 2, pp. 578–593, 2015.

[8] S. P. Berge and T. I. Fossen, "Robust control allocation of overactuated ships; experiments with a model ship," in *Proc. 4th IFAC Conf. Manoeuvring and Control of Marine Craft*, 1997, pp. 166–171.

[9] M. Rindaroey and T. A. Johansen, "Fuel Optimal Thrust Allocation in Dynamic Positioning," *IFAC Proceedings Volumes*, vol. 46, no. 33, pp. 43–48, 2013.

[10] A. Veksler, T. A. Johansen, and R. Skjetne, "Thrust allocation with power management functionality on dynamically positioned vessels," in *2012 American Control Conference (ACC)*. IEEE, 2012, pp. 1468–1475.

[11] W. Durham, "Constrained control allocation," *Journal of Guidance, Control, and Dynamics*, vol. 16, no. 4, pp. 717–725, 1993.

[12] ——, "Efficient, Near-Optimal Control Allocation," *Journal of Guidance, Control, and Dynamics*, vol. 22, no. 2, pp. 369–372, mar 1999.

[13] T. I. Fossen, *Guidance and Control of Ocean Vehicles*, 1994, vol. 32, no. 8.

[14] T. A. Johansen, I. Petersen, and O. Slupphaug, "Explicit sub-optimal linear quadratic regulation with state and input constraints," *Automatica*, vol. 38, no. 7, pp. 1099–1111, jul 2002.

[15] P. Tøndel, T. A. Johansen, and A. Bemporad, "An algorithm for multi-parametric quadratic programming and explicit MPC solutions," *Automatica*, vol. 39, no. 3, pp. 489–497, mar 2003.

[16] T. a. Johansen, T. I. Fossen, and S. P. Berge, "Constrained nonlinear control allocation with singularity avoidance using sequential quadratic programming," *Control Systems Technology, IEEE Transactions on*, vol. 12, no. 1, pp. 211–216, jan 2004.

[17] P. Tøndel, T. A. Johansen, and A. Bemporad, "Evaluation of piecewise affine control via binary search tree," *Automatica*, vol. 39, no. 5, pp. 945–950, may 2003.

[18] A. Veksler, T. A. Johansen, and S. Member, "Dynamic positioning with model predictive control," *IEEE Transactions on Control System Technology*, pp. 1–14, 2015.

[19] A. Sørensen, S. Sagatun, and T. Fossen, "Design of a dynamic positioning system using model-based control," *Control Engineering Practice*, vol. 4, no. 3, pp. 359–368, mar 1996.

[20] P. T. K. Fung and M. J. Grimble, "Dynamic Ship Positioning Using a Self-Tuning Kalman Filter," *IEEE Transactions on Automatic Control*, vol. 28, no. 3, pp. 339–350, mar 1983.

[21] S. Saelid, N. Jenssen, and J. Balchen, "Design and analysis of a dynamic positioning system based on Kalman filtering and optimal control," *IEEE Transactions on Automatic Control*, vol. 28, no. 3, pp. 331–339, mar 1983.

[22] T. I. Fossen, *Handbook of Marine Craft Hydrodynamics and Motion Control*. Chichester, UK: John Wiley & Sons, Ltd, apr 2011.

[23] A. J. Sørensen, "A survey of dynamic positioning control systems," pp. 123–136, 2011.
[24] R. G. Brown and P. Y. C. Hwang, *Introduction to Random Signals and Applied Kalman Filtering*, 4th ed. J. Wiley, 1992.
[25] T. I. Fossen and J. P. Strand, "Passive nonlinear observer design for ships using lyapunov methods: full-scale experiments with a supply vessel," *Automatica*, vol. 35, no. 1, pp. 3–16, 1999.
[26] B. Rokseth, S. Skjong, and E. Pedersen, "Modeling of Generic Offshore Vessel in Crane Operations With Focus on Strong Rigid Body Connections," *IEEE Journal of Oceanic Engineering*, vol. PP, no. 99, pp. 1–23, 2016.
[27] A. Richards, "Fast Model Predictive Control with soft constraints," *European Journal of Control*, vol. 25, pp. 51–59, 2015.
[28] B. Houska, H. J. Ferreau, and M. Diehl, "ACADO toolkit-An open-source framework for automatic control and dynamic optimization," *Optimal Control Applications and Methods*, vol. 32, no. 3, pp. 298–312, may 2011.
[29] H. J. Ferreau, C. Kirches, A. Potschka, H. G. Bock, and M. Diehl, "qpOASES: a parametric active-set algorithm for quadratic programming," *Mathematical Programming Computation*, vol. 6, no. 4, pp. 327–363, 2014.
[30] O. Faltinsen, *Sea Loads on Ships and Offshore Structures*. Cambridge University Press, 1993.
[31] S. Skjong and E. Pedersen, "On the Modeling and Control of Synchronous Generators for use in Marine Power Plants," *To be submitted*, 2017.

Stian Skjong received the MSc degree in marine technology from the Norwegian University of Science and Technology (NTNU) in 2014, and is currently working towards the Ph.D degree in marine technology, also at NTNU. His research interests include modeling, simulation and control of marine systems, nonlinear control and model based control designs, as well as co-simulation strategies for virtual prototyping of marine systems and operations.

Eilif Pedersen received the M.Sc. degree in marine engineering from the Norwegian Institute of Technology, Norway, in 1983. He has been with the Norwegian Marine Technology Research Institute, as a Senior Research Engineer until 1999 when he joined The Norwegian University of Science and Technology, as an Associate Professor. His areas of expertise are in the field of modeling methodology and simulation of dynamic multidisciplinary and mechatronic systems focusing on machinery system dynamics, internal combustion engines, vibrations, thermal- and hydraulic machines, fuel-cell system dynamics, and hybrid power plants for marine applications. He has held multiple positions within the university such as Vice Dean of Education at the Faculty of Engineering Science and Technology, Head of Master Programs in Marine Technology, Leader of the Research Group of Marine Systems and Head of Machinery Laboratory at the Department of Marine Technology. Currently he is the Leader of the Power Systems and Fuels work package at the Smart Maritime Centre for Research-Based Innovation.