Dynamic positioning with model predictive control

Aleksander Veksler[1], *Member, IEEE*, Tor Arne Johansen[1], *Senior Member, IEEE*,
Francesco Borrelli[2], *Senior Member, IEEE*, and Bjørnar Realfsen[3]

[1]Center for Autonomous Marine Operations and Systems, Department of Engineering Cybernetics (NTNU), Norway.
[2]Model Predictive Control Laboratory, University of California, Berkeley, CA 94720 USA.
[3]Kongsberg Maritime, Kongsberg, Norway.

Marine vessels with dynamic positioning capability are typically equipped with many enough thrusters to make them overactuated, and with satellite navigation and other sensors to determine their position, heading and velocity. An automatic control system is tasked with coordinating the thrusters to move the vessel in any desired direction and to counteract the environmental forces. The design of this control system is usually separated into several levels. First, a dynamic positioning (DP) control algorithm calculates the total force and moment of force that the thruster system should produce. Then, a thrust allocation (TA) algorithm coordinates the thrusters so that the resultant force they produce matches the request from the DP control algorithm. Unless significant heuristic modifications are made, the DP control algorithm has limited information about the thruster effects such as saturations and limited rate of rotation of variable-direction thrusters, as well as systemic effects such as singular thruster configurations. The control output produced with this control architecture is therefore not always optimal, and may result in a position loss that would not have occurred with a more sophisticated control algorithm. Recent advances in computer hardware and algorithms make it possible to consider model-predictive control algorithm (MPC) that combines positioning control and thrust allocation into a single algorithm, which theoretically should yield a near-optimal controller output. The presented work explores advantages and disadvantages of using model predictive control compared to the traditional algorithms.

***Index Terms*—Marine vehicles, position control, optimal control**

I. Introduction

A marine vessel is said to have dynamic positioning (DP) capability if it is able to maintain a predetermined position and heading automatically exclusively by means of thruster force [1]. DP is therefore an alternative, and sometimes a supplement to the more traditional solution of anchoring a ship to the seabed. The advantages of positioning a ship with the thrusters instead of anchoring it include:

- Immediate position acquiring and re-acquiring. A position setpoint change can usually be done with a setpoint change from the operator station, whereas a significant position change for an anchored vessel would require repositioning the anchors.
- Anchors can operate on depths of only up to about 500 meters. No such limitations are present with dynamic positioning.
- No risk of damage to seabed infrastructure and risers, which allows safe and flexible operation in crowded offshore production fields.
- Accurate control of position and heading.

The main disadvantages are that a ship has to be specifically equipped to operate in DP, and that dynamically positioned ships consume more power to stay in position, even though anchored vessels also have to expend energy to continuously adjust the tension in the mooring lines.

DP is usually installed on offshore service vessels, on drill rigs, and now increasingly on production platforms that are intended to operate on locations that are so deep that permanent attachment to the sea floor with e.g. mooring lines is impractical. Several thousand ships worldwide have dynamic positioning installed.

Depending on its function, a DP vessel that fails to keep its position may risk colliding with other vessels, endanger divers, interrupt operations and/or cause damage to equipment such as risers. Vessels that are designed to perform reliably typically have a high degree of redundancy in all critical systems, including power generation, thrusters and the computer system. Classification societies such as DNV GL and International Maritime Organization (IMO) issue standards that include safety regulations for the DP systems. For example, to be classified as a IMO Class 2, the DP system has to be designed with redundancies in the power distribution system, power generation, thruster system and many others; in particular, the thruster system must continue to be fully capable after failure of any single thruster.

The algorithm that coordinates the thrusters to keep a setpoint position and orientation is called the dynamic positioning algorithm. A commonality for most control algorithms that are available in the literature is that they separate the control task into two parts. First, a high-level motion control algorithm considers the current position and heading of the ship, and determines the total force and moment of force (together called "generalized force", further explained in Appendix B) that needs to be applied on the ship. Somewhat ambiguously, this algorithm is usually called the DP control algorithm. After the generalized force is calculated by the motion control algorithm it is passed as an input to a lower-level thrust allocation algorithm, which determines the directions and magnitudes of the forces that the individual thrusters should produce. The main goal of the thrust allocation algorithm is to ensure that the total generalized force that the thrusters generate matches

Corresponding author: Aleksander Veksler (email: aleksander.veksler@itk.ntnu.no).

the commanded output from the high-level motion control algorithm. The output from the thrust allocation algorithm is then sent to the local thruster controllers. The local controllers often have load rate limiting functionality to ensure that the load from the thrusters don't increase so fast that the power plant ends in blackout. A functionality to limit the thruster torque is usually also present.

Achieving the dynamic positioning task may be trivial if the environmental conditions are favorable, positional precision requirements are leisurely and the operator is not too concerned about costs such as fuel and wear-and-tear of the machinery. For the high-level motion controller, one can use three independent PID controllers one for each degree of freedom, and a simplistic thrust allocation algorithm as described in Subsection II-B1.

More advanced algorithms aim to have faster position acquiring and recovery, less rapid variation in the thruster commands, handling of variable-direction thrusters, better handling of thruster limitations, etc. Several well-functioning algorithms for the high-level motion control are known, many of them are described in [2, Subsection 12.2]. Also, in [3], the high-level motion control is implemented as an MPC algorithm, resulting in a controller that combines use of leisurely control effort as long as it is sufficient to keep the vessel within a predefined operational area, and more aggressive control effort when dynamic simulations show that the vessel would leave the operational area otherwise; the task of allocating the force order to the individual thrusters is in that implementation left with a classical thrust allocation algorithm. A similar MPC-based high-level motion controller was discussed in [4], although without simulating disturbances or including constraints on the position of the vessel [5], [6], [7].

Thrust allocation remains an active field of research. The most recent trends are towards integration and increased information passing between the thrust allocation and other systems on the ship. In [8], a thrust allocation method that balances how much the thrusters load the different parts of the power plant to reduce the NOx production of the power plant. Using thrust allocation to reduce the load variations on the power plant has been explored in e.g. [9], [10], [11]. Also notably, in [12] the local thruster control was modified to achieve the same purpose, thus bypassing the thrust allocation algorithm. A recent review of the state of the art thrust allocation is available in [13].

The separation into a high-level motion control algorithm and a thrust allocation algorithm has the advantage that it results in a segmentized software architecture. However, since the high-level motion control algorithm does not consider thruster effects such as saturations, asymmetric energy consumption, reconfiguration time (azimuths and rudders), the generalized force command it produces must necessarily be sub-optimal. Industrial implementations usually perform heuristic adaptions to reduce the impact of those suboptimalities, allowing information to pass back from the thrust allocation algorithm to the high-level motion control algorithm. To an extent, this blurs the separation between the algorithms. It usually results in performance that approaches optimal, but at the cost of a large engineering effort and complicated tuning.

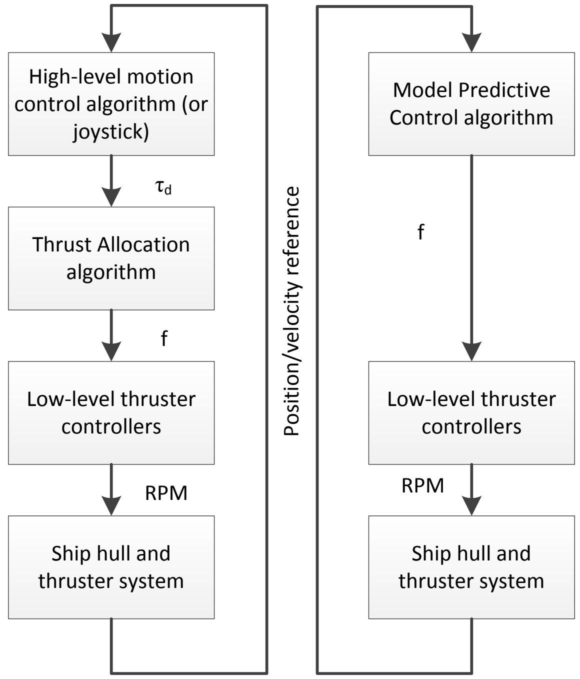

High-level motion control algorithm (or joystick)
τ_d
Thrust Allocation algorithm
f
Low-level thruster controllers
RPM
Ship hull and thruster system
Position/velocity reference
Model Predictive Control algorithm
f
Low-level thruster controllers
RPM
Ship hull and thruster system

Fig. 1: The architecture of the feedback loop from the position and velocity reference to the thruster commands for a ship equipped with DP. The classical architecture is to the left, the newly-suggested architecture is to the right.

A control strategy that simulates the future behavior of the system it controls, and based on the simulation results attempts to find a control output that makes the (simulated) system behave optimally is called "Model Predictive Control", or MPC; the optimal behavior is typically defined as the one that minimizes a specified metric, for example the energy consumption by the actuators, the average distance from the setpoint or a weighed combination thereof. In some cases, the output of the MPC can be calculated with an explicit algebraic expression, but often finding it requires a large number — sometimes millions — of possible scenarios to be evaluated. Introductory texts on MPC are available in e.g. [14], [15]. Continuing expansion of MPC to new applications is made possible by increasingly faster computer hardware, as well as more efficient and user-friendly software design tools [16], [17], [18], [19]. For example, [20], [21], [22], [23] explore control of a diesel-electric power plant with MPC, and MPC has also been used to realize dynamic control allocation [24], [25]. In many of those applications, MPC would earlier be considered to be unimplementable in real time, or too complex to design and maintain.

In light of these developments it is natural to consider controlling a ship by dynamic positioning with a model predictive control algorithm with a sufficiently long horizon, controlling the thrusters directly without a separate thrust allocation algorithm. This architecture is illustrated to the right in Figure 1. The expected advantages are:

- More consistent constraint handling. Implementing a high-level motion control algorithm that takes the dynamic thruster capabilities into account is possible with MPC.
- Planning ahead. Even with a reasonably short horizon an MPC can plan ahead maneuvers instead of going towards the goal in the most direct manner possible. Such maneuvers include moving the vessel towards the setpoint along a trajectory that is optimal and coordinated with respect to the saturation limits of the thrusters, as well as their direction and rate of turn where this is applicable. If the MPC is implemented with a sufficiently long horizon, it may be able to escape from local minima, such as situations where a bidirectional thrusters are acting in a direction where they are suboptimal.
- Simpler design and tuning. Since the model predictive control resolves a number of idiosyncrasies that require complex adaptions with the traditional architectures automatically, the engineering effort that is required to design the controller, is also expected to be smaller.

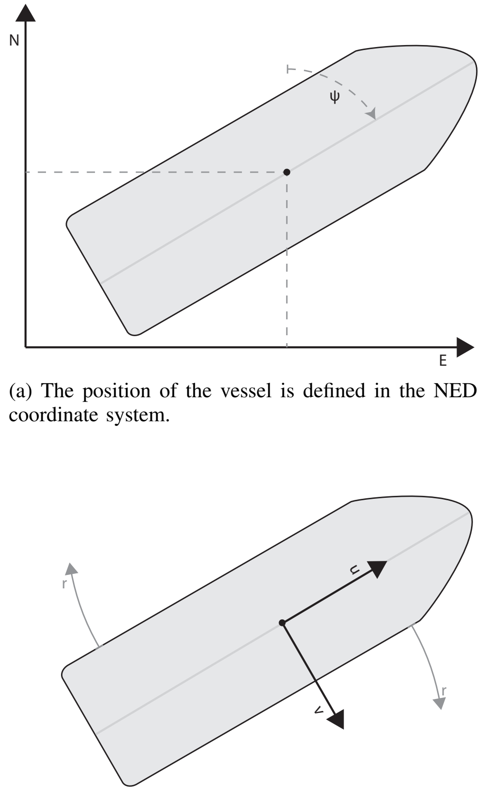
The advantages listed in the first two bullet points result mainly result in reduced power consumption and reduced position deviation in situations with limited thrust and power. The third one result in reduced development cost and faster time-to-market; the time spent configuring the DP system on-site may also be reduced, although it is usually possible to do most of the configuration work off-site with hardware-in-the-loop simulation [26].

The goal of this paper is to explore the advantages and implementability MPC through computer simulation to determine if the technology is viable for implementation. We chose to keep the formulation as simple as possible without adding any explicit stability-enforcing terms or constraints. The main reason for this is that a sufficiently large prediction horizon is sufficient to guarantee stability, e.g. [27]. Moreover, the system is not unstable, and simulations show no problems with stability. If it is necessary to reduce the prediction horizon to simplify computations in an industrial implementation, stability and performance might be influenced. It is well known how to re-design MPC with stability-enhancing cost terms and constraints, e.g. [15]. Potential practitioners should be aware of the limited scope of this work, as a practical implementation would have to handle additional complication such as model-plant mismatch in all parts of the model, imperfect position reference and wave filters. Practical DP codes tend for this reasons to become very large and remain proprietary.

The mathematical model that is used for forecasting is described in Section II, and the MPC problem is formulated and discussed in Section III. Its implementation as a computer program is discussed in Section IV. The proposed MPC controller was tested in simulation, and the results from four simulation scenarios are presented in Section V. A glossary of some of the more important terms can be found in Appendix B.

Abbreviation	Description
$\eta = \begin{bmatrix} N & E & \psi \end{bmatrix}^T \in \mathbb{R}^3$	Position and orientation of the vessel in an inertial frame of reference, in this case North-East-Down
$\nu = \begin{bmatrix} u & v & r \end{bmatrix}^T \in \mathbb{R}^3$	Velocity of the vessel in its own (body) frame of reference, expressed at the end point of the vector $\begin{bmatrix} N & E \end{bmatrix}^T$

TABLE I: Abbreviations that are used to describe the position and velocity of the vessel, as per convention from [28] and [2, especially p. 19].



(a) The position of the vessel is defined in the NED coordinate system.

(b) The velocity is defined in the body coordinate system.

Fig. 2: Coordinate systems.

II. Brief introduction to modeling of marine vessels

This section provides a basic explanation of the model that is used to implement the model predictive control in this work.

A. Geometry and kinematics

The investigation that is presented here deals with a ship that moves on the ocean surface at relatively low velocities. The roll and pitch motions of the vessel are neither monitored nor compensated. The mathematical model that is used to describe the system can therefore be kept reasonably simple by limiting it to the planar position and orientation of the vessel. A coordinate system is selected with the origin at the DP setpoint, x-axis pointing to North, y-axis pointing to East, and the (unused) z-axis pointing downwards per the right-hand rule. The orientation of the ship in the xy-plane is defined as clockwise rotation with the orientation with the bow pointing to the North as the zero reference.

The velocity of the vessel is usually described in its own frame of reference, in what is called the “body” coordinate system. The forward velocity u in the direction of the bow, sideways velocity v towards the starboard and the clockwise yaw rotation rate r. The abbreviations that are used to describe the position and the velocity of the vessel are presented in Table I and in Figure 2. The relationship between them is purely geometric, and is specified by

$$\dot{\eta} = \underbrace{\begin{bmatrix} \cos(\psi) & -\sin(\psi) & 0 \\ \sin(\psi) & \cos(\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix}}_{P(\psi)} \nu \tag{1}$$

Note that $P(\psi)P(\psi)^T = I$.

B. Dynamics

The forces and torques (taken together they are called “generalized force”[29]) from several physical sources act on the vessel. The resultant effect of those forces is equal to their vector sum; the same applies to the torques as long as they are expressed with the same pivot point. The control force is generated by the thrusters. Other forces that act on the ship include hydrodynamic drag, waves and wind. As mentioned above, the velocity and angular rotation speed are expressed in a frame of reference that is bound to the ship. This frame of reference may be rotating, which means that it is not inertial. The equations of motion in a rotating frame of reference normally have to include corrective terms for Coriolis and centripetal pseudo forces. If the rate of yaw rotation is modest however, these terms may reasonably be disregarded, as will be done in this treatment. For low speeds a linearization of the hydrodynamic drag is also reasonable.

Any acceleration of an object moving through a fluid requires that the surrounding fluid is also accelerated to move out of the way of the object. This creates a force on the object that is proportional to acceleration of the object. Mathematically, this effect is equivalent to an increase of the mass of the object [30, p. 567], and it is usually called “added mass” in hydrodynamic modeling. Unlike the physical mass, added mass is not symmetric, and for ships it is typically larger in the lateral direction than in the longitudinal direction. It can be shown that the resulting equations of motions can be represented in vector form as

$$M\dot{\nu} + D\nu = \tau + P^T(\psi)\tau_{env} \tag{2}$$

where $M \in \mathbb{R}^{3\times 3}$ is the generalized mass matrix and represents the sum of the physical mass and the hydrodynamic added mass. The drag approximation $D\nu$ is conventionally placed on the left side of this equation, changing the signs of the elements of D accordingly.

The thruster forces are represented by τ. The environmental forces that are not included in $M\dot{\nu}$ or $D\nu$ are collected in τ_{env}. Those are the forces due to current, high- and low-frequency components of the wind and wave forces, and the wind. The formulation in (2) implies that τ_{env} is expressed in the NED coordinate system, which is of course a matter of convenience.

Counteracting those forces doesn’t always require measuring them. The current force and the low-frequency components of the wave forces can be handled by the integral action in the DP control algorithm; it will also to some extent compensate for the actual thruster forces being different from what is modeled (ref. Subsection II-B2). Typically, it is not necessary to compensate for the high-frequency components of the wave forces, since they are essentially rock the boat back and forth. Those motions are usually discarded by a wave filter (ref Appendix B) before the position measurement is sent to the DP system. The wind forces are usually estimated with wind sensors; theoretically, this can be done fairly accurately, but the practitioners often encount complications due to the difficult geometry of typical ships and local variations in the wind speed.

1) Resultant thruster force calculations

A thruster that is located at r_i relative to the origin of a common coordinate system, generating force f_i at an angle α_i clockwise from the forward direction (ref Table II) will generate a generalized force

$$\tau_i = \begin{bmatrix} \cos(\alpha_i) \\ \sin(\alpha_i) \\ -l_{yi}\cos(\alpha_i) + l_{xi}\sin(\alpha_i) \end{bmatrix} f_i \tag{3}$$

The resultant generalized force from all the thrusters can be represented as

$$\tau = T(\alpha)f \tag{4}$$

where the columns of $T(\alpha) \in \mathbb{R}^{3\times Nthrusters}$ are of the form (3). A very simple thrust allocation algorithm can be implemented by Moore-Penrose pseudo-inverting $T(\alpha)$, calculating the force commands f per

Abbreviation	Description
$f \in \mathbb{R}^{Nthrusters}$	The thrust vector describing the forces produced by each thruster that is installed on the ship, with f_i being the force produced by thruster i.
$r_i = \begin{bmatrix} l_{i,x} & l_{i,y} \end{bmatrix}^T$	The location of the thruster device with index i
$\alpha \in \mathbb{R}^{Nthrusters}$	The directions in which the thrusters are pointing, with α_i being the direction of thruster i; $a_i = 0$ means that the thruster i is pointing backward and is generating the force forward unless reversed..
$\tau \in \mathbb{R}^3$	The resultant generalized force produced by all thrusters on the vessel
$T(\alpha) \in \mathbb{R}^{3\times Nthrusters}$	Thruster effect matrix. The linear operation $T(\alpha)f$ calculates and sums up the general forces produced by the individual thrusters.

TABLE II: Location and orientation of the thrusters, and the forces they produce on the ship.

$$f = \underbrace{\left(T(\alpha)T^T(\alpha)\right)^{-1} T^T(\alpha)}_{T^+(\alpha)} \tau \tag{5}$$

This algorithm does not consider thruster saturations or azimuth changes, so the algorithms in practical use are normally more advanced[13].

2) Relationship between generated force, RPM, and power consumption of the thrusters

Dimensional analysis of a propeller in free water (i.e. far from a ship hull or other obstructions and disturbances) combined with a few other hydrodynamical assumptions lead to a model where both the thrust and the torque produced by a propeller which is stationary in water are proportional to the square of the speed of rotation of the propeller [31, p. 145].

The power that is required to keep the propeller at a constant speed of rotation is the torque times the speed of rotation, which means that it is reasonable to assume that the power required to drive a propeller is proportional to the force it produces to the power of $3/2$. This approximation is used in many studies, including [13], [10], [11], while in the following we use quadratic force penalties instead of minimizing power consumption.

In this work, the actual force that is produced by the thrusters is assumed to be the controlled variable. This assumption implies that the local thruster controllers can accept a force setpoint [32].

C. Summary of the model

Motion of a vessel in DP can thus be approxiamted with equations (1), (2) and (3). Restating them for the sake of clarity,

$$\dot\eta = P(\psi)\nu \tag{6}$$

$$M\dot\nu + D\nu = \tau + P^T(\psi)\tau_{env} \tag{7}$$

$$\tau = T(\alpha)f \tag{8}$$

III. MPC FORMULATION

The following continuous-time numerical optimization formulation summarizes the discussion in the previous section, and is proposed as a basis for the model predictive control of the dynamical system:

$$J^* = \min_{f^+, f^-, \dot\alpha} \int_0^{T_e} \left\{ \left\|f^+\right\|^2_{Q_{f+}} + \left\|f^-\right\|^2_{Q_{f-}} + \left\|\dot\alpha\right\|^2_{Q_\alpha} + \right.$$

$$\left. + \left\|\nu\right\|^2_{Q_\nu} + \left\|\eta\right\|^2_{Q_\eta} + \left\| \int_0^{T_e} \eta dt \right\|^2_{Q_{\int \eta}} \right\} dt$$

$$+\text{similar terminal costs} \tag{9}$$

subject to

$$T(\alpha)f^+ - T(\alpha)f^- = \tau \tag{10}$$

$$0 \leq \left[\begin{array}{cc} f^+, & f^- \end{array}\right]^T \leq \left[\begin{array}{cc} \bar f, & \underline{f} \end{array}\right]^T \tag{11}$$

$$|\dot\alpha| \leq \bar{\dot\alpha} \tag{12}$$

$$M\dot\nu + D\nu = \tau - P(\psi)^T\tau_{env} \tag{13}$$

$$P(\psi)\nu = \dot\eta \tag{14}$$

$$\text{Initial conditions on } \eta, \nu, \alpha \tag{15}$$

$$\tau(T_e) = P(\psi)^T\tau_{env}(T_e) \tag{16}$$

Implementation of this problem as a computer algorithm is explained in Section IV. The thruster system constraint (10) is based on (4), but it is separated into a positive and a negative term to accomodate assymmetric thrusters. Asymmetric thrusters are designed for maximal efficiency in one direction, and are less efficient when running in reverse. The cost associated with running them in reverse is for this reason higher, and usually the maximal attainable thrust is lower.

The rate of change of the vector of azimuth angles for the variable-angle thrusters α is limited in (12) because some thrusters such as azumuth thrusters and rudderes cannot change their direction very quickly. The maximum rate of rotation varies, but is typically on the scale of 30 seconds for a 360 degree turn.

With traditional thrust allocation algorithms, a slack term is often necessary in (10) to make sure that the allocation problem is feasible even if a thrust command cannot be achieved. In this case, this term is not necessary, since τ represents the actual generalized force on the vessel, and not a setpoint command as with the traditional thrust algorithms.

The constraint (11) limits the force produced by the individual thrusters. Not all thrusters can produce force in negative direction, and some thrusters can produce more force in positive direction than in negative, so the values of $\overline{f}$ and $\underline{f}$ have to be set accordingly.

The constraints (13)–(14) are the dynamic and kinematic constraints of the ship, and are a vector form formulation of the Newtonian laws of motion for the system as discussed in Section II. Environmental generalized force τ_{env} represents the resultant of the various environmental forces such as wind, currents, waves, moorings, anchors, risers, cables (cable lay), pipes (pipe lay), hoses (offloading), sea ice, etc. Typically, the high-frequency component of the wave forces are filtered out of the model. The value of τ_{env} at the initial time can be estimated with a combination of direct measurment with e.g. wind sensors and model-based observers; the latter is mathematically equivalent to the integral action in a traditional

PID controller. The future values of τ_{env} can of course not be known, which is one of the factors that limit the forecasting power of the model. This will be further discussed in Subsection III-B.

The terminal costs represent how good the system state is at the end of the horizon. They can also be seen as a representation of the costs that will be incurred after the optimization horizon. The only terminal costs that were found to be necessary in the tested models are the cost associated with the the generalized position η and the cost associated with its integral.

Without the terminal constraint (16), the optimizer will turn off the thrusters at the end of the optimization horizon in the open-loop simulation. This way it will capitalize on the fact that the resulting drift-off lags after the thrusters turn off; this drift-off cost will thus fall after then end of the prediction horizon T_e and thus "outside" (9). The open-loop trajectory will thus be sub-optimal at the end of the horizon. The impact of this sub-optimality on the closed-loop may be reduced by extending the length of the horizon, which of course increases computational complexity.

The DP setpoint is set to the origin of the NED coordinate system without loss of generality.

A. *Turning the thrusters around*

A complication with the standard control architecture for dynamic positioning is that the thrust allocation problem is often non-convex. In practice, this means that a thruster may end up stuck in a direction where it either has to produce thrust in reverse of its optimal direction ($f^- > 0$), or, in case of unidirectional thrusters, a direction where it cannot produce any thrust that can be useful. A standard solution to this is having an exogenous algorithm evaluate the situation, and turn the thrusters around if this is considered beneficial. Other sources of non-singularity are forbidden sectors and the use of rudders [13], [7].

An MPC formulation with a horizon that is as long as the turn-around time for the thrusters could automatically consider if the thrusters should be turned or not. Care should be taken that either numerical solver can reliably find non-local minima, or, as in case with the test implementation in Section IV, that the numerical solver is provided with an approximate direction for where to look by means of an appropriate warm-start trajectory.

The transition of the thrusters incurs a short-term cost, and unless the horizon is significantly longer than the turn-around time, the majority of the benifit of turning the thrusters is seen in the (reduction of the) terminal costs. Care should therefore be taken during tuning to ensure that the terminal costs correctly reflect the long-term advantage of having a thruster point in the right direction.

B. *Modeling simplifications and predictability*

The MPC solver simulates the expected behavior of the vessel and calculates the thruster commands that result in a behavior of the ship that is optimal based on specified criteria. In general, the quality of MPC increases with the length of the prediction horizon. However, the stochastic nature of the marine environment limits the predictive power of any possible mathematical model. The predictive power is certainly not improved by the approximations that are made during the model design in Section II.

The constraint (13) requires an estimate of the environmental forces τ_{env} until the end of the simulation horizon T_e (more specifically, the part of the environmental forces that cannot be predicted and filtered by the wave filter, ref Appendix B). It is technically conceivable, but not practical, to use information from nearby vessels or a fleet of UAVs to map the wind field and thus predict most of the variations in the environmental forces some time ahead. Normally the environmental forces don't change a lot during the time used as the prediction horizon, so it may be assumed that the environmental force is constant and equal to its value at the beginning of the horizon. The latter can be calculated using an observer based on (2). The quality of any such estimate naturally degrades with increased length of T_e. It is therefore unreasonable to expect that the forecasting made by the model predictive control is a close approximation of the actual behavior of the ship.

For these reasons, extending the prediction horizon T_e beyond a few seconds may not necessarily be better than other heuristic adaptions.

IV. Implementation

In this section, the implementation of the continuous-time MPC formulation (9)–(16) as a computer program will be discussed.

The MPC problem formulation includes the continuous-time constraints (12)–(14) that describe the physical laws of the system. If the state variables α, ν and η and the control outputs f^+, f^- and $\dot{\alpha}$ satisfy those constraints, then they represent a possible trajectory of the system – in this case a possible scenario for the movement of the vessel. The constraint (15) ensures that this trajectory is in fact a possible future trajectory of the vessel by matching the initial values of α, ν and η to the current state of the physical system.

Every possible future trajectory has a cost associated with it which is described by (9). There are costs associated with the use of thruster force, rotating the thrusters, and with position and velocity deviations from the setpoint. *A global solution of this problem is a possible trajectory for the states and control outputs that minimizes (9).* Because of computational limitations, this cost is only evaluated for a limited time, until T_e.

The continuous-time formulation has to be discretized to be solved on a digital computer. In a discrete-time representation, the continuous-time trajectories of variables such as position of the ship and thruster forces are represented by a (finite) number of discrete samples; the continuous-time constraints should then be replaced with a finite number of equality constraints between the samples. For example, with a prediction horizon of 45 seconds and discretization interval of 0.5 seconds, there will be 90 discrete variables representing each (schalar) continuous variable. By doing this for each state and input, it is possible to represent (9)–(16) as a – typically very

Unit	Normalization variable	Numerical value
Length/Position	L	$76.20\,\mathrm{m}$
Force	Mg	$4.507 \cdot 10^7\,\mathrm{kg} \cdot {}^{m}/_{s^2}$
Moment	MgL	$3.4342 \cdot 10^9\,\mathrm{kg} \cdot \mathrm{m} \cdot {}^{m}/_{s^2}$

TABLE III: Relevant bis normalization constants [2, table 7.2].

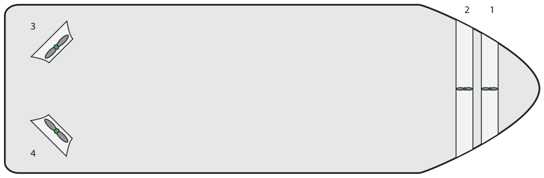


Fig. 3: Thruster layout of the simulated vessel, with two tunnel thrusters near the bow two rotable azimuth thrusters at the stern. The thruster numbers are shown.

large – numerical optimization problem; several algorithms are available to solve such problems, both open-source and commercial.

Software packages that are capable of handling discrete- or continuous-time MPC formulations also exist. Typically, they will also automatically hand off the resulting discrete-time numerical optimization problem to a solver. The package that was selected for this project is called BLOM, which stands for Berkeley Library for Optimization Modeling. It is desgined to assist in rapid implemention of model predictive control by providing a graphical interface to allow users to create optimization problems using Simulink. Using such package allows significant time savings compared to tranforming the model into a numerical optimization problem, and graphical design usually means faster and less error-prone implementation implementation.

BLOM has a capability to handle discretization automatically, but in this implementation the model was discretized “by hand” with Forward Euler discretization scheme.

V. Simulation results

The proposed thrust allocation algorithm was tested in simulation, on a model of SV Northern Clipper, featured in [33]. It is $76.20\,\mathrm{m}$ long, with a mass of $4.591 \cdot 10^6\,\mathrm{kg}$. It has four thrusters, with two tunnel thrusters near the bow and two azimuth thrusters at the stern. This configuration is illustrated in Figure 3. The maximal force for each thruster was set to ${}^1/_{60}$ of the ship’s dry weight. The turn-around time for the thrusters was set to relatively slow 60 seconds. All the thrusters are bi-directional; the tunnel thrusters in the bow are symmetric, while the azimuth thrusters use twice as much power to produce force in reverse compared to their normal direction. It was implemented in discrete-time in BLOM, using a forward Euler discretization with 0.5 seconds discretization interval. Four simulations are reported. The difference between them is that they have different starting conditions and different prediction horizons. Two of the scenarios also have increased relative heading priority, to better reflect how they will typically be used. The differences between the scenarios are summarized in Table IV.

The environmental forces were set to be a constant wind in the direction towards North. All numerical values were transfered to per-unit in bis system per Table III. The performance of the algorithm is compared to a baseline algorithm with a standard architecture with a separate DP control algorithm and a TA algorithm; it is based on elements from several available publications on the topic, and it is described in detail in Appendix A.

First scenario

The first simulation is a deliberate worst-case scenario for the classical controller. Both azimuth thrusters point towards the starboard, resulting in a singular thruster configuration. At the start, there is no way for the thruster system to create force in the longituidal direction where it is needed. The results are shown in Figures 4–7. The classical motion control immediately drives the thrusters to saturation, forcing them to act against one another without producing much force in total. The MPC-based controller does not command the thrusters to use significant forces until they have had the time to turn so that they can produce significant force in the direction where it is needed. In fact, the MPC controller *also turns the ship* to allow some use of the the bow thrusters, which would otherwise remain perpendicular to the direction where force is needed most and therefore almost useless. The maximal deviation is $0.41\,\mathrm{rad}$. Such behavior is desirable in many situations while in others it is essential to maintain the heading of the ship. In the second and the third configurations, the cost associated with deviation in yaw is set to be much larger, so the resulting wandering in yaw is much smaller.

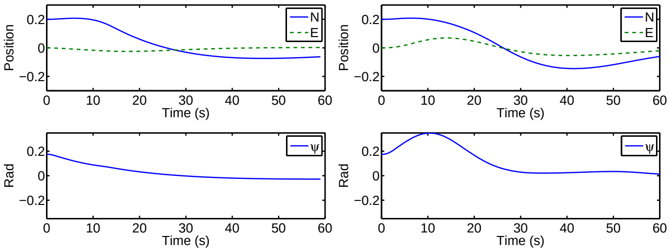
As can be seen in Figure 5, the MPC formulation leaves the system with both the azimuth thrusters pointing in the optimal direction, which is right against the environmental forces. With the classical formulation only one thruster, number 3, does that. In the latter situation it is optimal to use thruster 3 to produce most of the force that is required to hold the ship against the wind, while the other thrusters are only used to keep the ship from turning in yaw, which is needed because thruster 3 is slightly off-center. The thruster force asymptotically approaches 0.009 bis (it does not have quite enough time to fully converge within the time frame that is included in the figure), while with MPC the two thrusters converge to producing about 0.005 bis each. As mentioned in Subsection II-B2, the power consumption can reasonably be approximated to be proportional to the produced force to the power of ${}^3/_2$, so the power consumption for thruster 3 is approximately $\left(\frac{0.009}{0.005}\right)^{3/2}/2 = 1.21$ times greater than the power consumed by the two thrusters controlled by the MPC algorithm. The total force produced by the thruster system is in both cases very close to the environmental force.

Second scenario

The second simulation scenario was set up so that a thruster turnaround was advantageous assuming that the environmental forces remain constant, by initially pointing the azimuth thrusters in the same direction as the environmental force (they will thus initially run in reverse to act against the environmental force). The baseline controller ends up locking the azimuth thrusters in the negative direction, while the MPC-based one successfully turns them around, as can be seen in Figures 8–10. Unlike in the first simulation, the heading

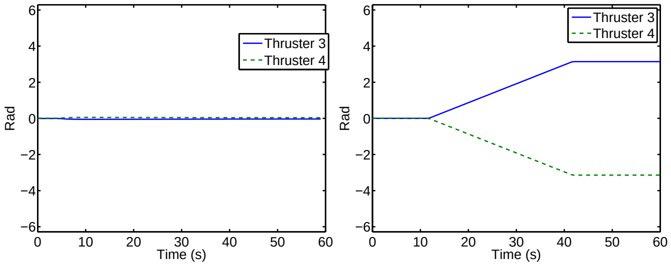
Scenario number	Starting position	Azimuth thrusters	MPC prediction horizon	Note
1	$\begin{bmatrix} 0.2 & 0 & 10^\circ \end{bmatrix}^T$	Both towards starboard	60 sec	
2	$\begin{bmatrix} 0.2 & 0 & 0^\circ \end{bmatrix}^T$	Both towards starboard	60 sec	Increased heading deviation penalty
3	$\begin{bmatrix} 0.2 & 0 & 0^\circ \end{bmatrix}^T$	45°torwards the center line	60 sec	Increased heading deviation penalty
4	$\begin{bmatrix} 0.2 & 0 & 10^\circ \end{bmatrix}^T$	Both towards starboard	10 sec	

TABLE IV: Simulation scenarios



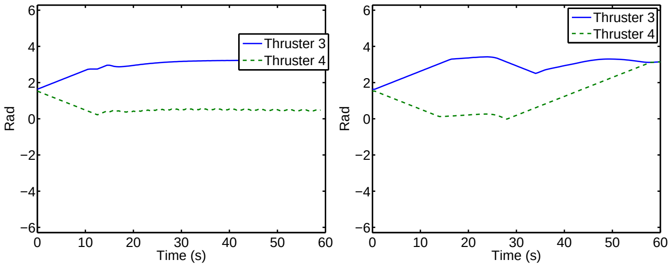
(a) Baseline (b) MPC

Fig. 4: *First scenario:* position of the vessel.



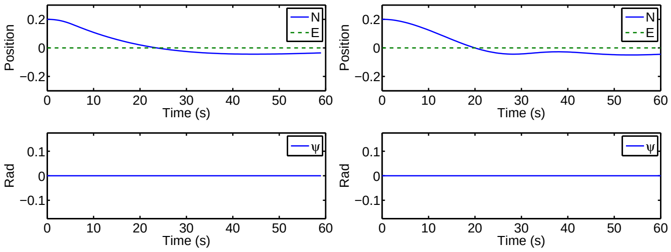
(a) Baseline (b) MPC

Fig. 8: *Second scenario:* angles of the azimuth thrusters.



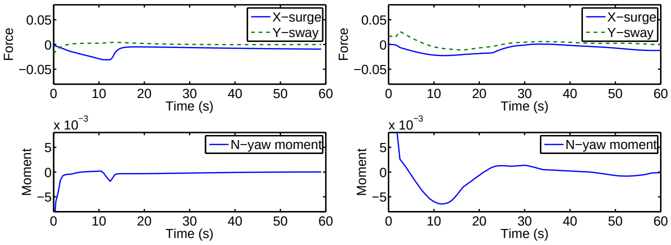
(a) Baseline (b) MPC

Fig. 5: *First scenario:* azimuth thruster angles.



(a) Baseline (b) MPC

Fig. 9: *Second scenario:* position of the ship.



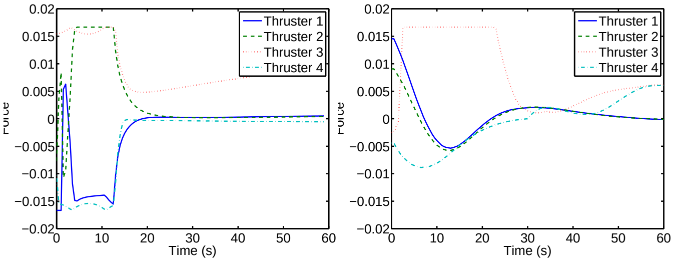
(a) Baseline (b) MPC

Fig. 6: *First scenario:* resultant forces.

remains close to constant in this case because of increased penalty for heading deviation. The vessel started with zero deviation in heading and along the West–East axis, resulting in motion of the vessel being mostly limited to one dimension. This is somewhat unnatural, but makes it easier to distinguish the various patterns in its movement.

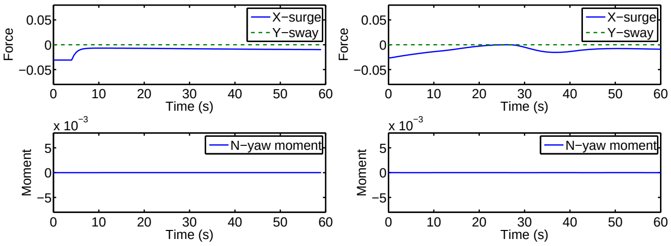
Third scenario

The third scenario is presented in Figures 11–14. It is more realistic, starting with the thrusters pointing towards the center line of the ship, exactly as illustrated in Figure 3. The resulting



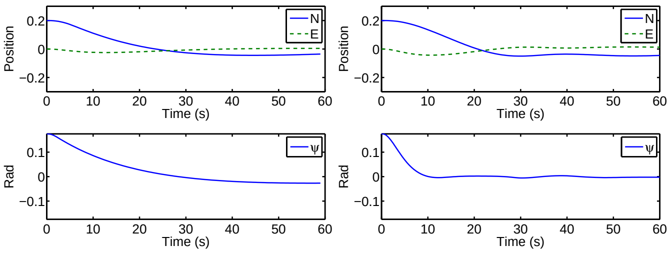
(a) Baseline (b) MPC

Fig. 7: *First scenario:* individual thruster forces.



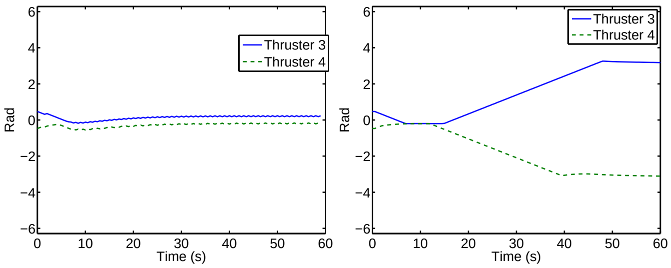
(a) Baseline (b) MPC

Fig. 10: *Second scenario:* resultant thruster forces.



(a) Baseline (b) MPC

Fig. 11: *Third scenario:* position of the vessel.



(a) Baseline (b) MPC

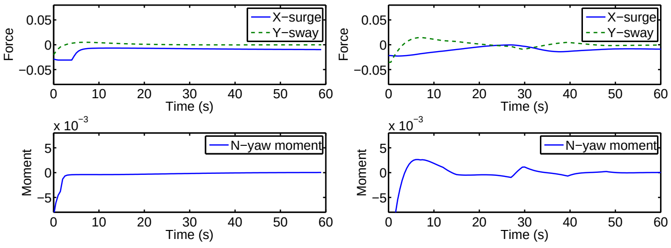
Fig. 12: *Third scenario:* azimuth thruster angles.

configuration is well-actuated, but, again, the azimuth thrusters point in a direction where they have to act in reverse to counteract the wind force. The MPC controller responds by first aligning the azimuth thrusters towards the center line to bring the vessel close to the setpoint as fast as possible, and when it is well underway about 10 seconds into the simulation, it starts the procedure of turning the azimuth thrusters around – a procedure it carefully coordinates with the position and velocity of the ship. The result is that at the end of the simulation the thrusters point in the right direction – with the classical controller they continue working in reverse, which requires usually requires much higher power consumption (twice of that in the forward direction with the parameters chosen for this simulation).

Fourth scenario

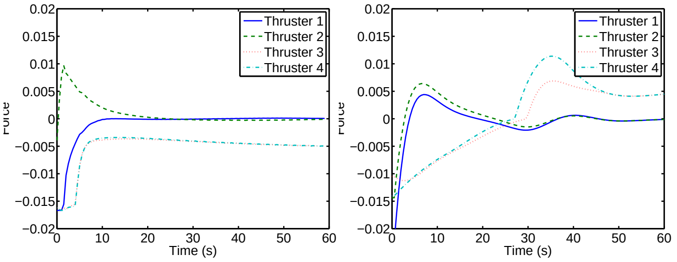
The fourth simulation is illustrated in Figures 15–18. Its starting configuration is similar to the first scenario, but the prediction horizon of the MPC controller was reduced to 10 seconds. With this prediction length, the MPC controller does not consider turning around the thrusters, since no advantages of doing that are seen within that period. When the transients settle, the forces from from the azimuth thrusters (thruster 3 and thruster 4) are close to the baseline in the nominal scenario. The MPC solver does not however atempt to create a force in an “impossible” direction, as is characteristic for classical controllers; also, the vessel turns itself slightly to allow the tunnel thrusters a better angle of attack, something a baseline controller would not consider.

The execution time on a laptop computer in this scenario is appoximately 2.7 seconds per iteration with cold start in the beginning of the simulation, and 0.2 seconds for the subsequent iterations which receive a reasonable initial trajectory guess; this is about five time faster than with the previous scenarios.



(a) Baseline (b) MPC

Fig. 13: *Third scenario:* resultant thruster forces.



(a) Baseline (b) MPC

Fig. 14: *Third scenario:* individual thruster forces.

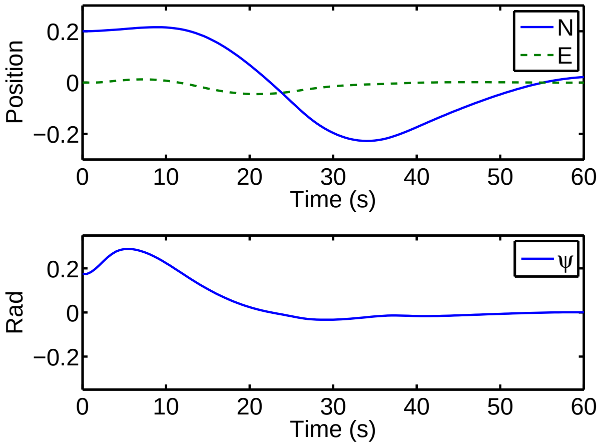


Fig. 15: *Fourth scenario:* position of the vessel

VI. Conclusion and future work

Implementing dynamical positioning with MPC appears to offer significant advantages compared to the current state-of-the-art. The MPC implementation is able to distribute force generation over a period of time and plan the motion of the vessel according to changes in configuration of the rotable thrusters; overall this results in less biasing (ref Appendix B, “Thruster Biasing”). Having large-scale model predictive control in real-time loops has its own challenges, mainly due to reliability, timely calculation of the results, and complexity of

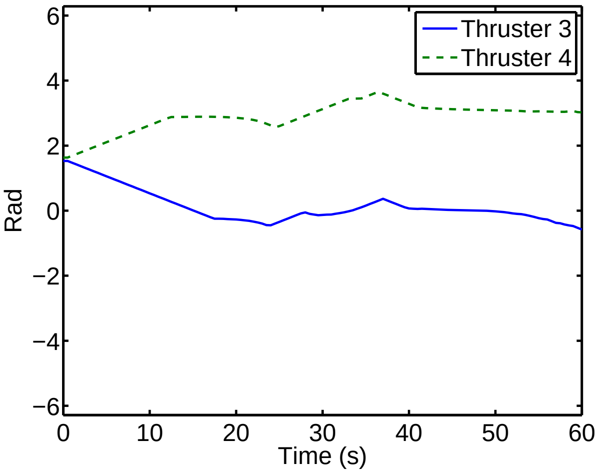


Fig. 16: *Fourth scenario:* azimuth thruster angles.

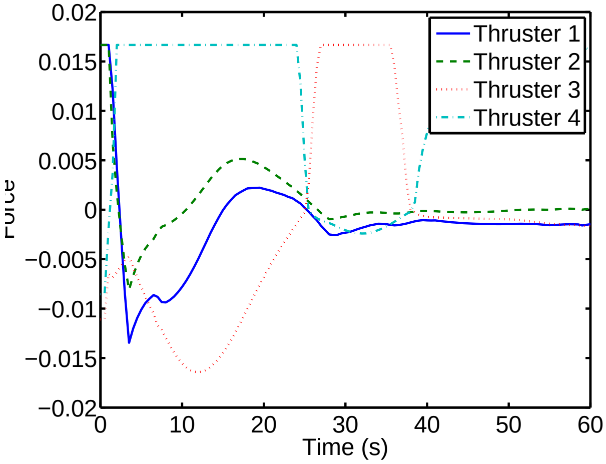


Fig. 17: *Fourth scenario:* individual thruster forces.

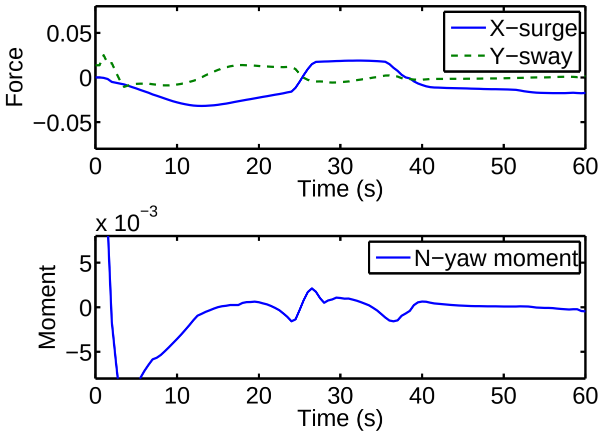


Fig. 18: *Fourth scenario:* resultant thruster forces.

the software implementation in embedded systems. Recent results are addressing this important issues from the perspectives of software tools (e.g. [34], [16]), computational efficiency [35], and dependability, e.g. [36]. Additionally, azimuth turning has to be implemented *ad hoc*, as with conventional thrust allocation. An MPC controller with a long enough horizon could theoretically dispose of this problem.

The execution time on the test computer suggests that this algorithm could feasibly be run in real-time on modern hardware. Convergence of nonlinear MPC is an active research area, and real-time guarantees on a complex model are difficult to make. A practical real-time implementation should have a classical DP implementation running as a safety fallback, in case the MPC solver fails to deliver a solution in time. Since the fallback algorithm is highly independent from the primary algorithm, the resulting architecture is in principle significantly more reliable than running a classical algorithm alone [37]. An MPC controller implementation with a shorter horizon should be investigated for possibility to run it real-time on current hardware.

VII. Acknowledgments

This work is partly sponsored by the Research Council of Norway, Kongsberg Maritime and DNV by the KMB project D2V, project number 210670 – SUP, and through the Centres of Excellence funding scheme, project number 223254 – AMOS. We greatly appreciate the discussions and advice contributed by Sergey Vichik and Tony Kelman of UC Berkeley, and Asgeir Sørensen of NTNU.

References

[1] *Rules for classification of ships*, DNV GL Std. Part 6 Chapter 7, Dynamic Positioning Systems, July 2013. [Online]. Available: https://exchange.dnv.com/publishing/RulesShip/2011-01/ts607.pdf

[2] T. I. Fossen, *Handbook of Marine Craft Hydrodynamics and Motion Control*. John Wiley & Sons, Ltd, April 2011.

[3] O. G. Hvamb, "A new concept for fuel tight DP control," in *Proc. Dynamic Positioning Conference*. Houston, Texas: Marine Technology Society, September 2001.

[4] H. Chen, L. Wan, F. Wang, and G. Zhang, "Model predictive controller design for the dynamic positioning system of a semi-submersible platform," *Journal of Marine Science and Application*, vol. 11, no. 3, pp. 361–367, 2012.

[5] S. P. Berge and T. I. Fossen, "Robust control allocation of overactuated ships; experiments with a model ship," in *in Preprints IFAC Conference on Maneuvering and Control of Marine Craft*, 1997.

[6] T. A. Johansen, T. I. Fossen, and S. P. Berge, "Constrained nonlinear control allocation with singularity avoidance using sequential quadratic programming," *IEEE Trans. Control Systems Technology*, vol. 12, pp. 211–216, 2004.

[7] T. A. Johansen, T. P. Fuglseth, P. Tndel, and T. I. Fossen, "Optimal constrained control allocation in marine surface vessels with rudders," *Control Engineering Practice*, vol. 16, no. 4, pp. 457 – 464, 2008.

[8] B. Realfsen, "Reducing NOx emission in DP2 and DP3 operations," in *Proc. Dynamic Positioning Conference*, 2009.

[9] E. Mathiesen, B. Realfsen, and M. Breivik, "Methods for reducing frequency and voltage variations on dp vessels," in *Proc. Dynamic Positioning Conference*, October 2012.

[10] A. Veksler, T. A. Johansen, and R. Skjetne, "Thrust allocation with power management functionality on dynamically positioned vessels," in *Proc. American Control Conf.*, 2012.

[11] ——, "Transient power control in dynamic positioning-governor feedforward and dynamic thrust allocation," in *9th IFAC Conference on Manoeuvring and Control of Marine Craft*, 2012.

[12] D. Radan, A. J. Srensen, A. K. dnanes, and T. A. Johansen, "Reducing power load fluctuations on ships using power redistribution control," *SNAME Journal of Marine Technology*, vol. 45, pp. 162–174, 2008.

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

[14] J. B. Rawlings, "Tutorial overview of model predictive control," *Control Systems, IEEE*, vol. 20, no. 3, pp. 38–52, 2000.

[15] D. Mayne, J. Rawlings, C. Rao, and P. Scokaert, "Constrained model predictive control: Stability and optimality," *Automatica*, vol. 36, no. 6, pp. 789 – 814, 2000. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0005109899002149

[16] B. Houska, H. J. Ferreau, and M. Diehl, "An auto-generated real-time iteration algorithm for nonlinear MPC in the microsecond range," *Automatica*, vol. 47, no. 10, pp. 2279 – 2285, 2011.

[17] D. Ariens, B. Houska, H. Ferreau, and F. Logist, "Acado: Toolkit for automatic control and dynamic optimization," Optimization in Engineering Center (OPTEC), http://www.acadotoolkit.org.

[18] Y. Wang and S. Boyd, "Fast model predictive control using online optimization," *Control Systems Technology, IEEE Transactions on*, vol. 18, no. 2, pp. 267–278, March 2010.

[19] D. K. M. Kufoalor, S. Richter, L. Imsland, T. A. Johansen, M. Morari, and G. O. Eikrem, "Embedded Model Predictive Control on a PLC Using a Primal-Dual First-Order Method for a Subsea Separation Process," in *Mediterranean Conference on Control and Automation*, Palermo, IT, June 2014.

[20] J. F. Hansen, "Modelling and control of marine power systems," Ph.D. dissertation, NTNU, Trondheim, Norway, 2000.

[21] T. I. B, "Dynamic model predictive control for load sharing in electric power plants for ships," Master's thesis, NTNU, 2012.

[22] T. I. B and T. A. Johansen, "Scenario based fault tolerant model predictive control for diesel-electric marine power plant," in *Proc. MTS/IEEE OCEANS*, 2013.

[23] A. Veksler, T. A. Johansen, E. Mathiesen, and R. Skjetne, "Governor principle for increased safety and economy on vessels with diesel-electric propulsion," in *Proc. European Control Conf.*, 2013.

[24] Y. Luo, A. Serrani, S. Yurkovich, D. B. Doman, and M. W. Oppenheimer, "Model predictive dynamic control allocation with actuator dynamics," in *Proc. American Control Conference, Boston*, 2004, pp. 1695–1700.

[25] M. Hanger, T. A. Johansen, G. K. Mykland, and A. Skullestad, "Dynamic model predictive control allocation using CVXGEN," in *9th IEEE International Conference on Control and Automation (ICCA'11), Santiago, Chile*, 2011.

[26] T. A. Johansen, A. J. Srensen, O. J. Nordahl, O. Mo, and T. I. Fossen, "Experiences from hardware-in-the-loop (HIL) testing of dynamic positioning and power management systems," in *OSV Singapore*, 2007.

[27] L. Grüne and J. Pannek, *Nonlinear Model Predictive Control: Theory and Algorithms*, 1st ed., ser. Communications and Control Engineering. Springer, 2011. [Online]. Available: http://www.springer.com/978-0-85729-500-2

[28] *Nomenclature for Treating the Motion of a Submerged Body Through a Fluid*, Technical and Research Bulletin No. 1-5, The Society of Naval Architects and Marine Engineers, Technical and Research Committee Std., 1950.

[29] H. Goldstein, C. P. Poole, and J. L. Safko, *Classical Mechanics*. Addison Wesley, June 2001.

[30] F. M. White, *Fluid Mech.* McGraw-Hill, 2003.

[31] J. D. van Manen and P. van Oossanen, "Propulsion," in *Principles of Naval Architecture*, E. V. Lewis, Ed. The Society of Naval Architects and Marine Engineers, 1988, ch. VI.

[32] L. Pivano, T. A. Johansen, and Ø. N. Smogeli, "A four-quadrant thrust estimation scheme for marine propellers: Theory and experiments," *Control Systems Technology, IEEE Transactions on*, vol. 17, no. 1, pp. 215–226, Jan 2009.

[33] T. I. Fossen, *Marine Control Systems*. Tapir Trykkeri, 2002.

[34] D. K. M. Kufoalor, B. J. T. Binder, H. J. Ferreau, L. Imsland, T. A. Johansen, and M. Diehl, "Automatic deployment of industrial embedded model predictive control using qpOASES," in *European Control Conference*, Linz, 2015.

[35] D. K. M. Kufoalor, S. Richter, L. Imsland, T. A. Johansen, M. M, and G. O. Eikrem, "Embedded model predictive control on a PLC using a primal-dual first-order method for a subsea separation process," in *22nd IEEE Mediterranean Conference on Control and Automation (MED'14)*, Palermo, Italy, 2014.

[36] T. Johansen, "Toward dependable embedded model predictive control," *Systems Journal, IEEE*, vol. PP, no. 99, pp. 1–12, 2014.

[37] A. Burns and A. J. Wellings, *Real-Time Systems and Programming Languages:*, 3rd ed. Addison Wesley, 2001.

[38] E. Ruth, "Propulsion control and thrust allocation on marine vessels," Ph.D. dissertation, NTNU, 2008.

[39] M. J. Grimble, R. J. Patton, and D. A. Wise, "The design of dynamic ship positioning control systems using stochastic optimal control theory," *Optimal Control Applications and Methods*, vol. 1, no. 2, pp. 167–202, 1980. [Online]. Available: http://dx.doi.org/10.1002/oca.4660010207

[40] V. Hassani, A. Sorensen, A. Pascoal, and A. Aguiar, "Multiple model adaptive wave filtering for dynamic positioning of marine vessels," in *American Control Conference (ACC), 2012*, June 2012, pp. 6222–6228.

Appendix A
Baseline controller

In order to evaluate the effectiveness of the the controller that is described in Section III, it has to be compared to an algorithm based on the standard control architecture. The dynamic positioning control algorithm that was used was a set of three PID controllers, with the velocity of the ship used as a derivative action.

The publications on thrust allocation algorithms that are currently available usually focus on solving only a few of the many facets of the thrust allocation problem at a time. Elements from several publications had to be collected into a single algorithm. The algorithm presented in [7] is selected as a starting point. The extension for the azimuth thrusters is based on [38], and the singularity avoidance is based on [6]. The algorithm represents the thruster forces in Cartesian coordinates.

The most important variables used in this section are introduced in Table V. Defining the control u_k for thruster k as

$$u_k = \begin{cases} [X_k, Y_k]^T & \text{if the thruster with index } k \text{ is rotable} \\ T_k & \text{otherwise} \end{cases} \tag{17}$$

and then defining the extended thrust vector u as $u = \left[\begin{array}{cccc} u_1^T & u_2^T & \dots & u_p^T \end{array}\right]^T \in \mathbb{R}^n$, where $n = 2p_r + p_f$ is the number of degrees of freedom available to the control system. The resultant generalized force from the thrusters is given by

$$\tau = Bu \tag{18}$$

where

$$B = \left[\begin{array}{cc} B_r, & B_f \end{array}\right] \tag{19}$$

with

$$B_r = \left[\begin{array}{ccccc} 1 & 0 & \cdots & 1 & 0 \\ 0 & 1 & \cdots & 0 & 1 \\ -l_{1,y} & l_{1,x} & \cdots & -l_{p_r,y} & l_{p_r,x} \end{array}\right]$$

and

$$B_f = \left[\begin{array}{ccc} \cos\alpha_{p_r+1} & \cdots & \cos\alpha_p \\ \sin\alpha_{p_r+1} & \cdots & \sin\alpha_p \\ l_{p_r+1} & \cdots & l_p \end{array}\right] \tag{20}$$

The matrix B describes the location and the orientation of all the thrusters on the vessel, and it is called the thruster configuration matrix.

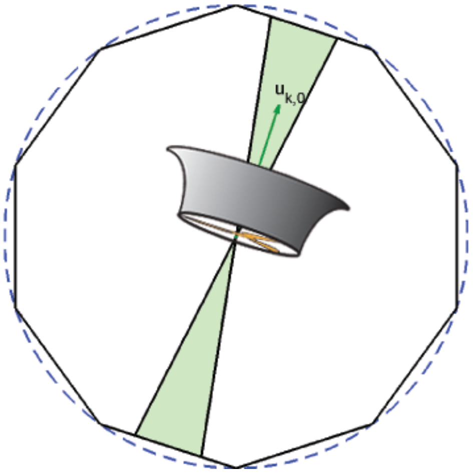


Fig. 19: An illustration of the constraints on the force that can be produced by an azimuth thruster. The force that was produced by that thruster in the previous iteration is shown as vector $u_{k,0}$. The dashed blue line is the saturation limit and the inscribed polygon is the linear approximation of that region. The light green sectors represent region within which a bidirectional thruster can produce force within the update interval. A unidirectional thruster can only produce force in one direction, which is the opposite of the direction in which it can push water.

A. Thruster saturation

As in [7], the constraints representing the thruster saturation for each rotable thruster is a polygonal approximation to the circular region, which is illustrated in Figure 19. This approximation can be done with arbitrary precision, and represented as a linear constraint in the form

$$A_k u_k \leq 1 \tag{21}$$

for a thruster with index k. Representing the saturation constraints for the non-rotable thrusters in the form (21) is straightforward.

B. Rotation rate constraint

Some rotable devices such as azimuth thrusters and rudders have a limited rate of rotation. This limitation is introduced as a constraint on the sector within which the force from such thrusters can be allocated in the iteration of the thrust allocation algorithm that is being calculated. The sector in which the rotable thruster with index k will be able to produce force is defined by angles $\alpha_{-,k}$ and $\alpha_{+,k}$ for each thruster k. Typically, $\alpha_{+,k} = \alpha_{k,0} + \Delta\alpha_{max}$ and $\alpha_{-,k} = \alpha_{k,0} - \Delta\alpha_{max}$, where $\alpha_{k,0}$ is the angle at which the thruster was in the previous thrust allocation and $\Delta\alpha_{max}$ is the maximal angle it can travel in the period between two allocations. If there are forbidden sectors that the thruster cannot enter, then the constraints on the allowed sector can be modified accordingly. As in [38, eqns (4.38), (4.39)], the sector constraint is represented as

$$\begin{bmatrix} \sin(\alpha_{-,k}) & -\cos(\alpha_{-,k}) \\ -\sin(\alpha_{+,k}) & \cos(\alpha_{+,k}) \end{bmatrix} u_k \leq 0 \tag{22}$$

C. Bidirectional thrusters

The sector constraint (22) automatically ensures that the corresponding thruster can only generate force in the direction that is opposite to where it is pointing. Some thrusters can reverse their direction by setting negative propeller speed or negative pitch. The constraint (22) can be modified to allow negative direction instead of positive by replacing it with

$$-\begin{bmatrix} \sin(\alpha_{-,k}) & -\cos(\alpha_{-,k}) \\ -\sin(\alpha_{+,k}) & \cos(\alpha_{+,k}) \end{bmatrix} u_k \leq 0 \tag{23}$$

For some thrusters either the sector constraints, the saturation constraints, or both are not symmetric, and have to be modified accordingly when the thrust is reversed.

A bidirectional thruster has to satisfy either (22) or (23), and in a special case it can satisfy both. In the simplest implementation, the trust allocation algorithm can be solved for all possible combinations of positive and negative thruster directions. For a configuration with $\overleftrightarrow{p_r}$ bidirectional thrusters this leads to $2^{\overleftrightarrow{p_r}}$ QP problems to be solved at each iteration.

Abbreviation	Description
p	The number of thruster devices on the vessel.
p_r, p_f	The number of rotable and the number of fixed-direction thruster devices.
$r_k = \left[l_{k,x}, l_{k,y}\right]^T$	The location of the thruster device with index k.
α_k	The angle of the thruster device with index k; α_k is constant for thruster devices with fixed direction and variable otherwise.
$T_k \in \mathbb{R}$	The force (magnitude) produced by device with index k.
X_k, Y_k, $N_k \in \mathbb{R}$	The force components in surge and sway, and the moment of force in yaw produced by the device with index k.
$\tau = \sum_k \begin{bmatrix} X_k, & Y_k & N_k \end{bmatrix}^T \in \mathbb{R}^3$	The resultant generalized force produced by all thrusters on the vessel.
τ_c	The generalized force order to the thrust allocation algorithm from the high-level motion control algorithm or from a joystick.
$u \in \mathbb{R}^{2p_r+p_f}$	The extended thrust vector.
u_0, $u_{k,0}$	The extended thrust vector from the previous iteration of the algorithm and the control u_k from the previous solution of the thrust allocation algorithm.

TABLE V: Nomenclature of the baseline thrust allocation algorithm.

D. Singularity avoidance

A situation where the thruster system that is constructed to be over-actuated cannot generate significant forces or moments in some directions without first rotating the thrusters is called a singularity situation. When in this situation, the vessel is vulnerable to e.g. rapid changes in the environmental forces, to which it may not be able to respond for a period of a few seconds.

A singularity situation may happen with a thruster system if its unrotable thrusters are not able to avoid this situation alone. Mathematically, the singularity situation is characterized by the thruster configuration matrix B becoming numerically close to being rank deficient.

The singularity avoidance technique that is used in this work is similar to [6]. The algorithm that is presented in that article introduces a penalty for proximity of the current configuration matrix to a singular configuration, as measured by

$$\frac{\varrho}{\epsilon + det(B_\alpha B_\alpha^T)} \tag{24}$$

where ϱ is a configurable cost parameter, ϵ is a small positive number that prevents the possibility of division by zero, and B_α is the configuration matrix in polar coordinates, with rows $B_{\alpha,i}$ defined as

$$B_{\alpha,i} = \begin{bmatrix} \cos\alpha_i \\ \sin\alpha_i \\ -l_{yi}\cos\alpha_i + l_{xi}\sin\alpha_i \end{bmatrix} \quad \forall i \in 1 \dots p \tag{25}$$

In this work, by defining

$$\tilde{u}_k = {}^{u_k}/_{\|u_k\|} = \begin{bmatrix} \cos(\alpha_k), & \sin(\alpha_k) \end{bmatrix}^T \tag{26}$$

it can be noted that

$$B_\alpha = \begin{bmatrix} B_{\alpha,rotable} & B_{\alpha,unrotable} \end{bmatrix} \tag{27}$$

with

$$B_{\alpha,rotable} = \begin{bmatrix} B_{1,1}\tilde{u}_1 + B_{1,2}\tilde{u}_2 & \cdots & B_{1,2p_r-1}\tilde{u}_{2p_r-1} + B_{1,2p_r}\tilde{u}_{2p_r} \\ B_{2,1}\tilde{u}_1 + B_{2,2}\tilde{u}_2 & \cdots & B_{2,2p_r-1}\tilde{u}_{2p_r-1} + B_{2,2p_r}\tilde{u}_{2p_r} \\ B_{3,1}\tilde{u}_1 + B_{3,2}\tilde{u}_2 & \cdots & B_{3,2p_r-1}\tilde{u}_{3p_r-1} + B_{3,2p_r}\tilde{u}_{2p_r} \end{bmatrix} \tag{28}$$

$$B_{\alpha,unrotable} = \begin{bmatrix} B_{1,2p_r+1}\tilde{u}_{2p_r+1} & \cdots & B_{1,n}\tilde{u}_n \\ B_{2,2p_r+1}\tilde{u}_{2p_r+1} & \cdots & B_{2,n}\tilde{u}_n \\ B_{3,2p_r+1}\tilde{u}_{2p_r+1} & \cdots & B_{3,n}\tilde{u}_n \end{bmatrix} \tag{29}$$

The quantity $\sqrt{det(B_\alpha B_\alpha^T)}$ is the volume of the 3-dimensional parallelepiped that is spanned by the row vectors of the matrix B_α, which approaches zero as B_α approaches a rank-deficiency. Thus, the quantity

$$J_{sing}(u) = \frac{\varrho}{\epsilon + det\left(B_\alpha \cdot B_\alpha^T\right)} \tag{30}$$

is a good penalty function for approaching the singularity condition. By linear approximation at the current azimuths the change in cost will be

$$\bar{J}_{sing}(u) = J_{sing}(u_0) + \left.\frac{dJ_{sing}(u)}{du}\right|_{u_o}^T \cdot (u - u_0) \tag{31}$$

Equation (26) assumes that $u_k \neq \mathbf{0}$ even if the magnitude of the thruster force is zero. This can be ensured in an implementation by replacing that magnitude with a nonzero but physically insignificant number, so that the vector u_k always carries the information about the direction of the thruster.

E. Optimization problem formulation

Collecting the cost and constraint terms from the previous sections and writing it as a standard QP thrust allocation formulation yields

$$\begin{aligned} \min_{u,s} \quad & J(s,u) && (32) \\ \text{Subject to} \quad & && \\ \tau_c \quad &= \quad Bu + s && (33) \\ \pm\begin{bmatrix} \sin(\alpha_{-,k}) & -\cos(\alpha_{-,k}) \\ -\sin(\alpha_{+,k}) & \cos(\alpha_{+,k}) \end{bmatrix} u_k \quad &\leq \quad 0 && (34) \\ & \forall k \leq p_r && \\ A_k u_k \quad &\leq \quad 1 \forall k && (35) \end{aligned}$$

where

$$J(s,u) = u^T H u + (u - u_0)^T M (u - u_0) + s^T Q s + \left.\frac{dJ_{sing}(u)}{du}\right|_{u_0} \cdot u \tag{36}$$

with positive semidefinite cost matrices H, M and Q of appropriate dimension. For rotable thrusters which are only capable of producing force in the positive direction, the constraint (34) has to be satisfied with a positive sign. For thrusters which are capable of producing thrust in both positive and negative direction, the constraint (34) has to be satisfied with either positive or negative sign. An instance of the optimization problem can be solved with every possible combination of positive and negative constraints for each applicable k; normally the one with the lowest optimal cost $J^*(s,u)$ should be selected.

The slack vector s is present to ensure that the optimization problem does not become infeasible even if the thruster system is physically unable to produce the generalized force order τ_c. The weight matrix Q should be selected such that the optimal value of s is small unless the problem without s would be infeasible.

The cost function (36) also includes a quadratic approximation to the power consumption in the thrusters ($u^T H u$), a penalty for variations in the extended thrust vector that is intended to reduce wear-and-tear in the thrusters $\left((u - u_0)^T M (u - u_0)\right)$, and the part of the linearized singularity avoidance penalty (31) that is not constant in u.

The output from the optimization problem can be trivially converted to desired thruster forces and angles when applicable (except when u_k for a rotable thruster is 0, then the previous angle can be used). Mapping from the desired thruster force to the RPM setpoint for the frequency converter that feeds that thruster can be done with a model-based controller e.g. [32] or the inverse thrust characteristic.

Appendix B Glossary

Dynamic Positioning System, DP

An automatic system that maintains position of a ship or vessel using its thrusters, in presence of disturbances such as wind, waves and current. In the context of control algorithm design, the DP is an algorithm which assesses the position of the vessel based on various instruments, calculates how far it is from a desired position setpoint selected by the operator, and based on that decides the thruster force needed to get to the desired position.

Thrust allocation algorithm, TA

An algorithm that takes as input an order for the total force and moment that the thrusters should enact upon a ship, and calculates what forces the individual thrusters should produce so that resultant force and moment on the ship becomes as ordered. A more general concept of *control allocation* is applicable to any fully- or overactuated vehicle [13].

Generalized force

A generalization of the Newtonian force concept for systems that are described in generalized coordinates[29]. In the context of 3DOF surface ship modeling, generalized force is a three-dimensional vector consisting of the two-dimensional force vector and the torque acting on a chosen pivot point.

Thruster

Any unit capable of producing controlled thrust on the vessel. For some thrusters, the direction in which they produce thrust can be controlled, while others can only produce thrust in a fixed direction. The main types of thrusters used for dynamic positioning are:

- **Tunnel thrusters.** Typically located at the bow of the ship, tunnel thrusters consist of a "tunnel" through the hull of this ship, through which a propeller can push water, typically in either direction.
- **Azimuth thrusters** have their propellers mounted on a rotable assembly. This allows the azimuth thrusters to change their direction freely, with turn-around times on the scale of one minute. On may ships, the azimuth thrusters are not allowed to push water in certain directions, such as against other thrusters or at other critical machinery. This type of thruster is typically located at the aft of the ship, or beneath the columns of semi-submersible rigs.
- **Ruddered or unruddered propellers** are classical propellers, typically located at the aft of the ship and pointing backwards. They are often powered mechanically through a drive shaft. A rudder is most effective when a propeller pushes water pass its surface, which is typically behind the propeller. Rudders are therefore often installed behind propellers to allow generating a sideways force at the location of the rudder, which normally also creates a yaw moment on the ship.

Thruster biasing

Deliberately increasing the power consumption in the thrusters without changing the total produced force and moment on the ship, effectively forcing the thrusters to push against one another.

For a given azimuth and rudder angle vector α, the combined force vector and angular momentum produced by the thrusters is

$$\tau = B(\alpha)f \tag{37}$$

and is a linear combination of the forces f generated by the individual thrusters. If there are four or more thrusters on board the ship, then the matrix $B(\alpha)$ is guaranteed to have a non-trivial null space F_0. Additionally, if f^* is a strict global minimizer of the power consumption for a given τ, then for any $f_0 \in F_0 \setminus \mathbf{0}$ the power consumption for $f^* + f_0$ will be higher than for f^*, with the resultant generalized force remaining the same. Therefore, biasing can always be achieved as long as there are at least four non-saturated thrusters available for the purpose. Fewer than four thrusters are sufficient for configurations in which the columns of the matrix $B(\alpha)$ are not independent.

Wave filter

Waves characteristically induce a cyclic high-frequency "back-and-forth" motion on ships. Compensating for the wave-induced motions would demand large power expenditure and increased wear-and-tear. For this reason, the cyclic wave-induced motions of the vessel are usually allowed to run their course, without interference from the dynamic positioning system. This is implemented by passing the position reference through a wave filter algorithm, which calculates what the position of the vessel would have been without the cyclic wave-induced motions. Several methods for implementing such filter has been proposed, see e.g. [2], [39], [40].