Cartesian thrust allocation algorithm with variable direction thrusters, turn rate limits and singularity avoidance

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

***Abstract*— The literature on thrust allocation algorithms that is currently available usually focuses on solving only a few of the many facets of the thrust allocation problem at a time. This paper presents a unified thrust allocation algorithm that solves most of the challenges that are faced by the practitioners in one algorithm. This includes controlling thrusters that can change the direction of the generated thrust slowly and/or reverse the direction of the generated thrust, minimizing the power consumption and wear-and-tear in the thrusters, and handling thruster saturations. When rotable thrusters are present, a functionality to avoid driving the thruster system into singular configurations should normally be included. This functionality requires significant numerical calculations for each iteration of the thrust allocation algorithm. In the presented work those calculations were written in explicit form using a symbolic processor, translated to ANSI C and compiled. This technique was demonstrated to provide acceptable real-time performance.**

I. Introduction

Designing control systems for ships in dynamic positioning (DP) is subject to a long-lasting research effort, with practical implementations available since early sixties [1], [2]. The more recent efforts were directed towards designing algorithms that systematically resolve complications such as coordination of rotable thrusters [3], [4], control of ruddered propellers [5], power consumption optimization [6] and other power management-related issues [7], [8], [9], [10], sector constraints, and other issues that were earlier resolved heuristically. An introductory textbook is available in [11] and a recent review of the topic is available in [12]. These developments allow DP operations with better safety, economy, positioning precision and reliability, making it possible for the DP-equipped vessels to perform more complex tasks in deeper waters [13] and on arctic latitudes [14], [15].

DP algorithms are usually divided into several parts, one of which is the thrust allocation (TA) algorithm. The task of the TA algorithm is to receive a command telling it how much force and moment of force (jointly called "generalized force" in this context) the thruster system should produce, and generate commands down to the individual thrusters to ensure that the resultant force and moment of force on the ship matches the command that was received by the TA.

This paper proposes a TA algorithm that combines and integrates the elements from earlier contributions. This algorithm is able to control the most common thruster types used on vessels with DP, including azimuth thrusters, tunnel thrusters, ruddered propellers and Voith Schneider propellers, and it has a functionality for singularity avoidance. This algorithm is based on [5] and represents the thruster forces in Cartesian coordinates, allowing a consistent representation of the thruster forces between different types of thrusters. The extension for the azimuth thrusters is based on [16], and the singularity avoidance functionality is implemented by placing a singularity proximity penalty in the cost function of the optimization problem, similar to [3]. This work contributes a theoretical and a practical examination of the singularity avoidance functionality implemented as mentioned above.

II. Thrust allocation algorithm

The TA algorithm is stated as a numerical optimization problem in Subsection II-F, and Subsections II-A to II-E discuss the cost and constraint terms of this optimization problem. The most important variables used in this paper are introduced in Table I.

A. Resultant force calculation

Let the control u_k for thruster k be defined as

$$u_k = \begin{cases} [X_k, Y_k]^T & \text{if the thruster with index } k \text{ is rotable} \\ T_k & \text{otherwise} \end{cases} \tag{1}$$

and then define the extended thrust vector u as $u = \begin{bmatrix} u_1^T & u_2^T & \dots & u_p^T \end{bmatrix}^T \in \mathbb{R}^n$, where $n = 2p_r + p_f$ is the number of degrees of freedom available to the control system. The resultant generalized force from the thruster system can then be calculated per

$$\tau = Bu \tag{2}$$

where

$$B = \begin{bmatrix} B_r, & B_f \end{bmatrix} \tag{3}$$

with

*This work was supported by the RCN Strategic University Program on Control, Information and Communication Systems for Environmental and Safety Critical Systems

[1]Center for Autonomous Marine Operations and Systems, Department of Engineering Cybernetics (NTNU), Trondheim, Norway aleksander. veksler@itk.ntnu.no, tor.arne.johansen@itk.ntnu.no

[2]Department of Mechanical Engineering, University of California, Berkeley, CA 94720-1740, USA borrelli@berkeley.edu

[3]Kongsberg Maritime, Kongsberg, Norway. bjornar.realfsen@km.kongsberg.com

$$B_r = \begin{bmatrix} 1 & 0 & \cdots & 1 & 0 \\ 0 & 1 & \cdots & 0 & 1 \\ -l_{1,y} & l_{1,x} & \cdots & -l_{p_r,y} & l_{p_r,x} \end{bmatrix}$$

$$B_f = \begin{bmatrix} \cos\alpha_{p_r+1} & \cdots & \cos\alpha_p \\ \sin\alpha_{p_r+1} & \cdots & \sin\alpha_p \\ l_{p_r+1} & \cdots & l_p \end{bmatrix}$$

The matrix B describes the location and the orientation of all the thrusters on the vessel, and it is called the thruster configuration matrix. It is constant during normal DP operations.

B. Thruster saturation

As in [5], the constraints representing the thruster saturation for each rotable thruster are a polygonal approximation to the circular region, which is illustrated in Figure 1. This approximation can be done with arbitrary precision, and represented as a linear constraint in the form

$$A_k u_k \leq 1 \tag{4}$$

for a thruster with index k. The illustration shows an inscribed polygonal approximation, which (except in the special cases) prevents the thrusters from delivering their full capacity. If this is unacceptable it can be avoided by using a circumscribed polygonal approximation, accepting a small mismatch in the generated force instead. Alternatively, a modification that avoids this mismatch is described in [16, section 4.4.2].

Representing the saturation constraints for the non-rotable thrusters in the form (4) is straightforward.

C. Rotation rate constraint

Some rotable devices such as azimuths and rudders have a limited rate of rotation. This limitation is represented in the algorithm 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. Let angles $\alpha_{-,k}$ and $\alpha_{+,k}$ specify the sector in which the rotable thruster with index k will be able to produce force. 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 [16, eqns (4.38), (4.39)], the sector constraint is represented as

$$\begin{bmatrix} \sin(\alpha_{-,k}) & -\cos(\alpha_{-,k}) \\ -\sin(\alpha_{+,k}) & \cos(\alpha_{+,k}) \end{bmatrix} u_k \leq 0 \tag{5}$$

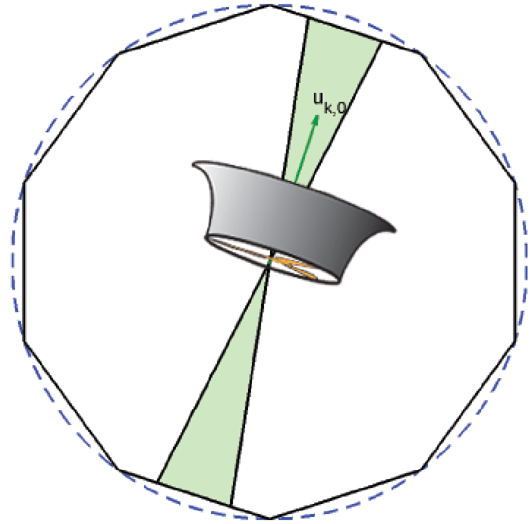

Fig. 1. An illustration of the constraints on the force that can be produced by an azimuth thruster. The force that was produced by that thruster in the previous iteration is shown as vector $u_{k,0}$. The dashed blue line is the saturation limit and the inscribed polygon is the linear approximation to 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.

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.
$r_k = [l_{k,x}, l_{k,y}]^T$	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 1. Nomenclature.

D. Bidirectional thrusters

The sector constraint (5) 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 (5) can be modified to allow negative direction instead of positive by replacing it with

$$-\begin{bmatrix} \sin(\alpha_{-,k}) & -\cos(\alpha_{-,k}) \\ -\sin(\alpha_{+,k}) & \cos(\alpha_{+,k}) \end{bmatrix} u_k \leq 0 \tag{6}$$

Both the sector constraints and the saturation constraints may be different when the thruster operates in reverse, and the constraints have to be modified accordingly.

A bidirectional thruster has to satisfy either (5) or (6), and in a special case it can satisfy both. In the simplest implementation, the thrust 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.

E. Singularity avoidance

A situation in which 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 polar thruster configuration matrix B_α becoming numerically close to being rank deficient; unlike B, B_α changes with changing thruster angles. Its rows $B_{\alpha,i}$ are defined as

$$B_{\alpha,i} = \begin{bmatrix} \cos\alpha_i \\ \sin\alpha_i \\ -l_{yi}\cos\alpha_i + l_{xi}\sin\alpha_i \end{bmatrix} \quad \forall i \in 1 \dots p \tag{7}$$

The singularity avoidance technique that is used in this work is similar to [3]. It introduces a penalty for proximity of the current polar configuration matrix to a singular configuration, as measured by

$$\frac{\varrho}{\epsilon + det(B_\alpha B_\alpha^T)} \tag{8}$$

where ϱ is a configurable cost parameter, ϵ is a small positive number that prevents the possibility of division by zero.

By defining

$$\tilde{u}_k = {}^{u_k}/_{\|u_k\|} = \begin{bmatrix} \cos(\alpha_k), & \sin(\alpha_k) \end{bmatrix}^T \tag{9}$$

it can be noted that

$$B_\alpha = \begin{bmatrix} B_{\alpha,rotable} & B_{\alpha,unrotable} \end{bmatrix} \tag{10}$$

with

$$B_{\alpha,rotable} = \begin{bmatrix} B_{1,1}\tilde{u}_1 + B_{1,2}\tilde{u}_2 & \cdots & B_{1,2p_r-1}\tilde{u}_{2p_r-1} + B_{1,2p_r}\tilde{u}_{2p_r} \\ B_{2,1}\tilde{u}_1 + B_{2,2}\tilde{u}_2 & \cdots & B_{2,2p_r-1}\tilde{u}_{2p_r-1} + B_{2,2p_r}\tilde{u}_{2p_r} \\ B_{3,1}\tilde{u}_1 + B_{3,2}\tilde{u}_2 & \cdots & B_{3,2p_r-1}\tilde{u}_{3p_r-1} + B_{3,2p_r}\tilde{u}_{2p_r} \end{bmatrix} \tag{11}$$

$$B_{\alpha,unrotable} = \begin{bmatrix} B_{1,2p_r+1}\tilde{u}_{2p_r+1} & \cdots & B_{1,n}\tilde{u}_n \\ B_{2,2p_r+1}\tilde{u}_{2p_r+1} & \cdots & B_{2,n}\tilde{u}_n \\ B_{3,2p_r+1}\tilde{u}_{2p_r+1} & \cdots & B_{3,n}\tilde{u}_n \end{bmatrix} \tag{12}$$

According to Corollary 1 in the Appendix, 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

$$J_{sing}(u) = \frac{\varrho}{\epsilon + det\left(B_\alpha \cdot B_\alpha^T\right)} \tag{13}$$

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

$$\bar{J}_{sing}(u) = J_{sing}(u_0) + \left.\frac{dJ_{sing}(u)}{du}\right|_{u_o}^T \cdot (u - u_0) \tag{14}$$

Equation (9) 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.

F. Optimization problem formulation

Collecting the cost and constraint terms from the previous sections and writing them down in a standard QP thrust allocation formulation yields

$$\min_{u,s} \quad J(s,u) \tag{15}$$

Subject to

$$\tau_c = Bu + s \tag{16}$$

$$\pm\begin{bmatrix} \sin(\alpha_{-,k}) & -\cos(\alpha_{-,k}) \\ -\sin(\alpha_{+,k}) & \cos(\alpha_{+,k}) \end{bmatrix} u_k \leq 0 \quad \forall k \leq p_r \tag{17}$$

$$A_k u_k \leq 1 \forall k \tag{18}$$

where

$$J(s,u) = u^T H u + (u - u_0)^T M (u - u_0) + s^T Q s + \left.\frac{dJ_{sing}(u)}{du}\right|_{u_0} \cdot u \tag{19}$$

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 (17) has to be satisfied with a positive sign. For thrusters which are capable of producing thrust in both positive and negative direction, the constraint (17) 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 (19) also includes a quadratic approximation to the power consumption in the thrusters ($u^T Hu$), 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 (14) 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. [17] or the inverse thrust characteristic.

III. Thrust allocation logic

Ideally, repeated iterations of the optimization problem (15)–(17) will converge the azimuth thrusters to an orientation that is optimal to counteract the environmental forces, possibly with some adjustments due to the singularity avoidance functionality. This does not happen in all situations. Detecting those situations, and implementation of other functionality is often implemented by computer logic that adds or modifies constraints and parameters within the numeric optimization framework. This includes:

- **Turning around bi-directional azimuth** thrusters that are running in reverse over extended periods of time. Although an electrical thruster can quickly revert direction, its propeller is in many cases optimized to push water in one specific direction. Pushing it in the opposite direction is energetically inefficient and also reduces the maximum capacity of the thruster. Turning a thruster around involves having that thruster point in a suboptimal direction over the time interval it takes to turn it around, incurring a significant short-term cost. Taking the decision to turn the thruster effectively means guessing how much the thruster would otherwise be pointing in the wrong direction in the future, how much it would affect efficiency of the said thruster and if those losses are enough to justify incurring the short-term cost of turning the thruster around. The exact answer depends on random wind variations, how the weather and the sea state continue to evolve and future decisions of the operator. This means that any decision would be inherently heuristic. A simple resolution to this issue is to turn around a thruster that has been running in reverse over a given period of time, or leave it to the operator to issue a reversal request. A more complex algorithm would make a statistical model of the environmental forces and even the operator, similar to applications in the automotive industry such as [18].
- **Crossing the forbidden sectors**. A thruster may not be allowed to push water in certain directions, e.g. into other thrusters, divers, or sensitive equipment. In order to pass a forbidden sector, a thruster may have to be set to zero RPM, which of course leads to a short-term loss of optimality. A heuristic algorithm should consider if this loss of optimality is worth the possible gain from having the thruster point in a more optimal direction in the long run.
- Using the thrust allocation algorithm to **improve the dynamics of the load on the power plant** through additional constraints and cost terms, as is done in [7], [8]. In many cases the power plant is separated into multiple segments. An efficient load distribution between the segments may lead to large reductions in fuel consumption. In [6], this is achieved by modifying the cost parameter H in (19) between the iterations of the algorithm depending on how the specific fuel consumption of the generators changes with their load.
- **Allowing the operator additional liberty** with regards to how he or she operates the ship. Operational aspects that are beyond the scope and indeed beyond the concern of the control algorithm designer may be managed by allowing the DP system to control only some of the thrusters,while keeping the rest under manual control or turning them off. In other circumstances, the operator may wish to fix the direction of some of the rotable thrusters, which corresponds to one additional constraint per thruster in the thrust allocation algorithm. Thrust allocation logic usually keeps track of which thrusters are ready to be used, which ones are running, and which ones are actually in use by the dynamic positioning system. Those three parameters may be combined in a number of ways.

IV. Simulation study

Fig. 2. The thruster layout of the simulated vessel.

The proposed thrust allocation algorithm was tested in a simulation, on a model of SV Northern Clipper, featured in [19]. The ship is 76.20 meters long, with a mass of $4.591 \cdot 10^6$ kg. It has four thrusters, with two tunnel thrusters near the bow and two azimuth thrusters at the stern. This layout is illustrated in Figure 2. The maximal force for each thruster was set to $^1/_{60}$ of the ship's dry weight. The dynamic positioning control algorithm that was used was a set of three PID controllers.

An explicit symbolic expression for the derivatives of the singularity avoidance cost (13) with respect to u was formed using MuPad. The expression is very long, in fact fairly measured by kilobytes. However, profiling demonstrated that on a laptop computer it takes 16 milliseconds to evaluate this expression as a Matlab function handle, and 0.4 milliseconds to evaluate it when it was compiled as a C function. This expression is a function of the extended thrust vector from the previous iteration (u_0) so it only has to be evaluated once for each iteration of the thrust allocation algorithm. The complexity of this expression is therefore not a hindrance against real-time implementation of the thrust allocation algorithm.

The system was implemented in Matlab/Simulink with Bis normalization system per e.g. [11, Section 7.2.5]. The simulated ship starts at $1/5$ of the ship's length away from the setpoint location, 10 degrees off the setpoint heading. The simulation results are shown in Figures 3–7. The starting configuration is singular, with all the thrusters pointing towards the starboard. In the first few seconds this leads to the thrusters being driven to saturation, as can be seen in Figure 4, while a significant deviation from the commanded generalized force is nevertheless observed (Figure 7). This situation illustrates the vulnerability that singular configurations represent – the thruster system may not be physically able to comply with the generalized force order. This problem is most severe in situations in which the direction of the generalized force order may change abruptly.

In steady-state conditions without the singularity avoidance term, it is optimal to have all the rotable thrusters point in the direction of the environmental load. The simulated vessel has only two fixed thrusters. They are located close to each other and they point in the same direction. So for this ship, any configuration in which the rotable thrusters point in the same direction as well (not necessarily the same direction as the fixed thrusters) is singular. However, because of the singularity avoidance term, after a transition – which involves rotation of the azimuth thrusters and positioning of the vessel at the setpoint – the azimuth thrusters converge to 185 and 24 degrees, leading to a slight bias with 20 degrees difference between the azimuth angles and therefore a nonsingular configuration.

Thruster 4 ends up working in reverse and is nearly unused at the end of the simulation. This inefficiency should be addressed by the thrust allocation logic.

V. Conclusion and Future Work

A workable thrust allocation algorithm that is a good basis for a practical implementation has been presented and tested in simulation. It is the assertion of the authors that this algorithm can be used on realistic vessels without significant modifications.

VI. Acknowledgments

This work is partly sponsored by the Research Council of Norway by the KMB project D2V, project number 210670, and through the Centres of Excellence funding scheme, project number 223254 – AMOS. Valuable comments that led to significant improvements of parts of the paper were contributed by Torstein Bø of NTNU and Tony Kelman of UC Berkeley.

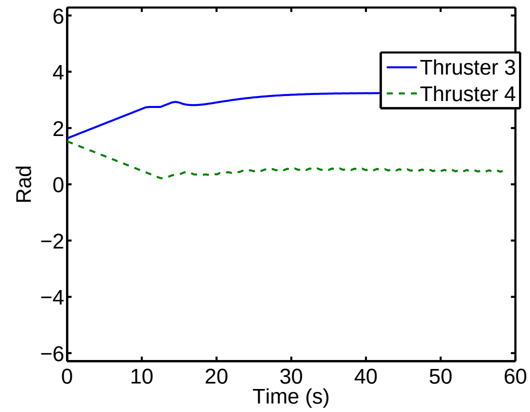

Fig. 3. Angles of the azimuth thrusters

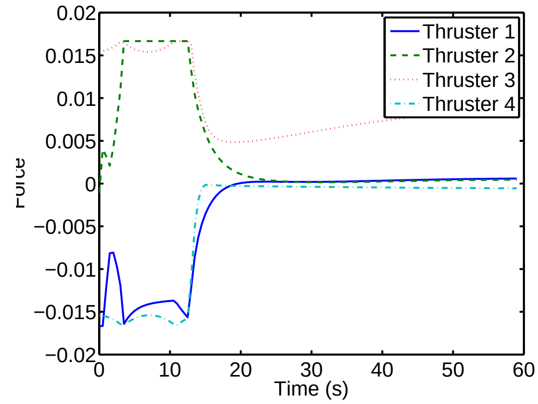

Fig. 4. Forces from the individual thrusters

Appendix

Theorem 1: For a matrix $A \in \mathbb{R}^{N \times M}$ with $N > M$, the volume of the M-dimensional parallelepiped that is spanned by the column vectors of A is given by

$$VolP(A) = \sqrt{det\,(A^T A)} = \prod_{i=1}^{M} S_{ii} \tag{20}$$

where S_{ii} is the i-th singular value of A.

Proof:

By singular volume decomposition, $A = USV^T$; the rows of the matrix $H = SV^T \in \mathbb{R}^{N \times M}$ consist of the M columns of V, multiplied by the respective elements of S, with the remaining $N - M$ rows of H being zero. The columns of V are orthonormal, so the volume of the parallelepiped that is spanned by the first M rows of H is $\mathcal{V} = S_{11}S_{22}\ldots S_{MM}$. Defining $H' \in \mathbb{R}^{M \times M}$ as the matrix consisting of those first M rows, notice that since H' is

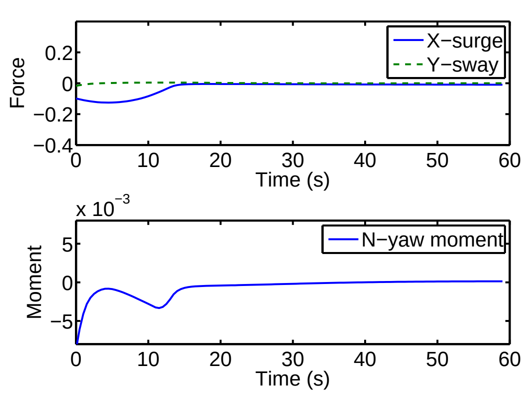

Fig. 5. Commanded generalized force

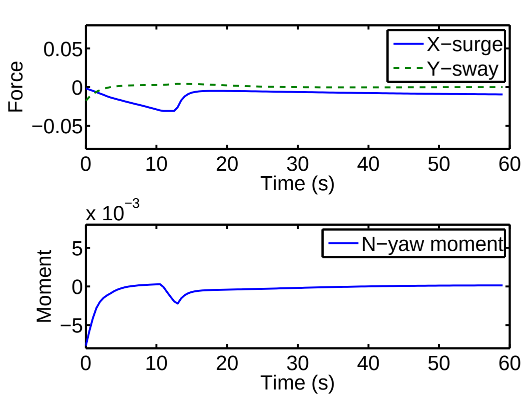

Fig. 6. Produced resultant generalized force

square, $VolP(H') = det(H')$. The volume that is spanned by the column vectors of H' is therefore also $\mathcal{V}$. The column vectors of H are identical to H', so its volume is also $\mathcal{V}$. Pre-multiplying H with the orthogonal matrix S does not affect lengths or relative angles of the column vectors of M, and the volume of the parallelepiped must also stay the same. Therefore $VolP(A) = \mathcal{V}$.

Examining (20),

$$\begin{aligned}\sqrt{det\left(A^TA\right)} &= \sqrt{det(VS^TU^T \cdot USV^T)} \\ &= \sqrt{det(V)det(S^TS)det(V^T)} \\ &= S_{11}S_{22}\dots S_{MM} = \mathcal{V}\end{aligned} \tag{21}$$

Therefore, $VolP(A) = \sqrt{det\left(A^TA\right)}$.

■

Corollary 1: For a matrix $A \in \mathbb{R}^{N\times M}$ with $N < M$, the volume $VolP(A)$ of the N-dimensional parallelepiped that is spanned by the row vectors of A is given by

$$VolP(A) = \sqrt{det\left(AA^T\right)} \tag{22}$$

Proof:

Let $A' = A^T$. Per Theorem 1, $VolP(A) = VolP(A') = \sqrt{det(A'A'^T)} = \sqrt{det(A^TA)}$.

■

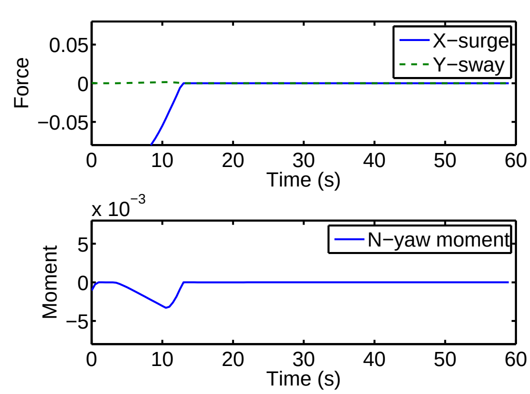

Fig. 7. The deviation between the produced and the commanded generalized force

REFERENCES

[1] In notes and comments, "Mohole trials yield samples of deep-water sediments," *New Scientist*, vol. 227, p. 730, March 1961.

[2] W. NIerenberg and M. Peterson, "Deep sea drilling project technical report no. 3," Scripps Institution of Oceanography, UCSD, Tech. Rep., december 1971. [Online]. Available: http://www.deepseadrilling.org/trepts/TRNOTE_03.PDF

[3] 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.

[4] O. J. Sørdalen, "Optimal thrust allocation for marine vessels," *Control Engineering Practice*, vol. 5, no. 9, pp. 1223 – 1231, 1997.

[5] T. A. Johansen, T. P. Fuglseth, P. Tøndel, 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.

[6] M. Rindarøy and T. A. Johansen, "Fuel optimal thrust allocation in dynamic positioning," in *9th IFAC Conference on Control Applications in Marine Systems*, Osaka, Japan, 2013.

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

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

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

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

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

[13] M. S. R. Costa and G. B. Machado, "Analyzing petrobras DP incidents," in *MTS Dynamic Positioning Conference*, Houston, 2006.

[14] Ø. K. Kjerstad, I. Metrikin, S. Løset, and R. Skjetne, "Experimental and phenomenological investigation of dynamic positioning in managed ice," *Cold Regions Sci. and Tech.*, 2014, in review.

[15] D. H. Nguyen, D. T. Nguyen, S. T. Quek, and A. J. S¿rensen, "Position-moored drilling vessel in level ice by control of riser end angles," *Cold Regions Science and Technology*, vol. 66, pp. 65–74, 2011.

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

[17] 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.

[18] T. Lin, E. Tseng, and F. Borrelli, "Modeling driver behavior during complex maneuvers," in *American Control Conference*, 2013.

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