Adv Syst Sci Appl 2017; **3**:1–8
Published online at http://ijassa.ipu.ru/ojs/ijassa/article/view/502

AUV Thrust Allocation with Variable Constraints

V.V. Kostenko[1], A.Yu. Tolstonogov[1,2*]

[1] *Institute for Marine Technology Problems FEB RAS, Sukhanova st. 5a, Vladivostok, 690091, Russia*
[2] *Far Eastern Federal University, Sukhanova st. 8, Vladivostok, 690091, Russia*

Abstract: Modern multipurpose Autonomous Underwater Vehicles (AUVs) represent the next generation of robotic systems with new technological tasks faced by researchers. One method of extending the functionality of the vehicle is installing tunnel thrusters in addition to the stern propulsion system. Thereby the vehicle becomes able to undertake both survey-style missions and low speed interactions with the environment. But the efficiency of the tunnel thruster depends strongly on the vehicle's velocity, due to hydrodynamic aspects. The usual solution to this problem uses different control models and thrust allocation methods for these types of mission. A unified approach to thrust allocation is presented in this paper. The approach is based on solving the allocation problem by quadratic programming with variable constraints, depending on the velocity of the AUV and the thrusters' angle of attack. The well-known implementation of active set method was used to model the proposed allocation method.

Keywords: AUV, propulsion system, thruster hydrodynamics, control allocation problem, quadratic programming, interior point method, active set method

1. INTRODUCTION

The design of a control algorithm for an underwater vehicle is often divided into several levels (Fig. 1.1) [1]. First, a high-level motion control algorithm is designed to compute a vector of virtual unbounded inputs to the vehicle $\boldsymbol{f_c}$ (eq. (1.1)) from the target and current vehicle states and the control type.

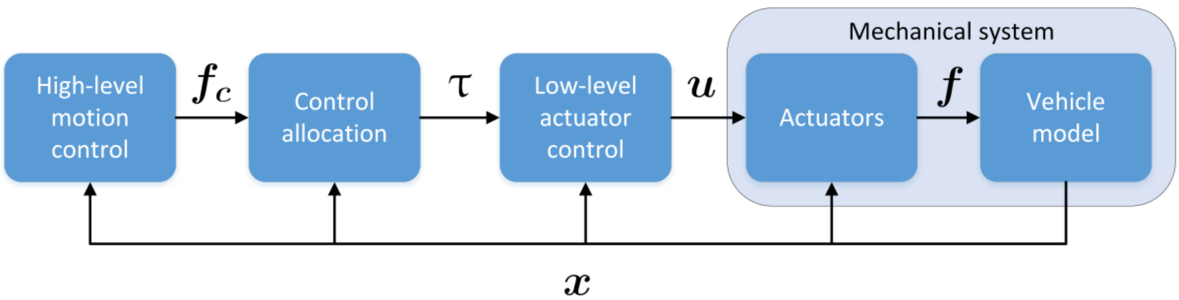


Fig. 1.1. Control system structure. $\boldsymbol{f_c}$ is the vector of commanded inputs, $\boldsymbol{f}$ is the virtual vector of inputs, $\boldsymbol{\tau}$ is the vector of allocated thrusts, $\mathbf{u}$ is the vector of low-level actuator inputs.

*Corresponding author: tolstonogov.anton@gmail.com

$$\begin{aligned} \dot{\boldsymbol{x}}(t) &= A(t)\boldsymbol{x} + B(t)\boldsymbol{f} \\ \boldsymbol{y}(t) &= C(t)\boldsymbol{x} \end{aligned} \tag{1.1}$$

Here, $\boldsymbol{x(t)} \in \mathbb{R}^n$ is the system state vector, t is time, $\boldsymbol{y}(t)$ is the vector of outputs, $\boldsymbol{f} \in \mathbb{R}^m$ is the vector of virtual inputs, which should be equal to the command vector $\boldsymbol{f_c}$ of the high level motion algorithm, and $A(t), B(t), C(t)$ are the coefficient matrices of the mechanical system. The virtual inputs are usually chosen to have a number of forces and torques that equals the number of degrees of freedom, m, that the motion control system wants to control.

Second, the control allocation algorithm is designed in order to map the vector of virtual input forces and torques $\boldsymbol{f_c}$ to individual thruster forces $\boldsymbol{\tau}$ such that the total forces and torques generated by all thrusters $\boldsymbol{f}$ amount to the commanded virtual input $\boldsymbol{f_c}$.

$$\boldsymbol{f} = B(\boldsymbol{x}, t)\boldsymbol{\tau} \tag{1.2}$$

Here, $B(\boldsymbol{x}, t)$ is the thruster configuration matrix. It contains the location and orientation of all thrusters, in the vehicle body-fixed frame. $\boldsymbol{\tau} \in \mathbb{A} \subset \mathbb{R}^p$ is the vector of actuator thrusts generated by the vehicle propulsion system, $\mathbb{A}$ are constraints from the saturation of the thrusters or other physical constraints, and p is the number of actuators.

Third, there is a separate high-frequency low-level controller for each actuator, that controls the desired thrust τ_i by a low-level control input u_i. For each actuator,

$$\tau_i = h_i(\boldsymbol{x}, t, u_i) \tag{1.3}$$

where h is a function, τ_i is the thrust of actuator i, and $u_i \in \mathbb{U} \subset \mathbb{R}$ is the low-level control input of actuator i. Usually, the effector model is linear in u:

$$\tau_i = h_i(\boldsymbol{x}, t, u_i) = G(\boldsymbol{x}, t)u_i \tag{1.4}$$

but the relation between the actuator thrust and the low-level control input is not considered in this article. We assume that each low-level controller maintains the desired thrust τ_i with sufficient quality.

This modularity allows the high-level motion control algorithm to be designed without detailed knowledge about the vehicle propulsion system. In addition to coordinating the effects of the different thrusters in the system, issues such as thruster/fault tolerance, redundancy, and control constraints are typically handled within the control allocation module. In the case of an over-actuated propulsion system, when the number of thrusters is greater than the number of degrees of freedom (DOF) controlled by the vehicle ($p > m$), the control allocation module solves an optimization problem to achieve a minimal power consumption of the propulsion system.

There are different approaches to the high-level motion control of a vehicle. The PID (proportional-integral-derivative) is a widespread control method, but new methods such as linear quadratic Gaussian (LQG) [2], the $\mathcal{H}_\infty$ control method [3], or fuzzy logic control, are being rapidly developed. An interesting method is Model Predictive Control [4], which solves a suboptimal motion control problem at each iteration, with the ability to combine the first and second levels of the motion, with the consequent increase in the computational complexity.

In this paper, the problem of high-level control is assumed to be solved and the only the problem of optimal thrust allocation in the case of an over-actuated vehicle is considered. This problem is well studied. There is a survey devoted to this problem [1]. Different approaches to this problem, including linear iterative approach as well as the quadratic programming approach satisfying optimality criteria, are treated in that survey. There are also papers published more recently [5–7].

These papers devoted to control allocation are focused on reducing the computational complexity of the optimal thrust allocation, but there is no mention of the fact that the thruster constraints can drift dynamically due to thruster hydrodynamics. A new approach, taking into account the vehicle speed and the thrusters' angle of attack, is proposed in the present article.

This approach allows controlling both survey-style missions and low speed interactions with the environment for a vehicle equipped with tunnel thrusters.

2. PROBLEM FORMULATION

Let the vector of virtual inputs computed by the high-level motion control or vehicle operator be denoted by the generalized force vector $\boldsymbol{f} = (f_x, f_y, f_z, m_x, m_y, m_z)^T$, here f_i is the projection of the force onto the axis i (i equals to the x, y and z axis) and m_i is the projection of the torque onto the axis i in the body-fixed coordinate frame. The body-fixed reference frame [8] is used in this work. The x axis is directed along the longitudinal vehicle axis from the vehicle's stern to the fore, the y axis is directed along the latitudinal vehicle to the starboard, and the z axis completes the frame to a right-handed coordinate system. Assume that the system is equipped with p thrusters with control thrust $\tau_i (i = 1, \ldots, p)$. This leads to the following relation between $\boldsymbol{\tau} = (\tau_1, \tau_2, \ldots, \tau_p)$ and the virtual inputs $\boldsymbol{f}$:

$$B\boldsymbol{\tau} = \boldsymbol{f} \tag{2.5}$$

According to [9], the optimization problem can be written as

$$\begin{gathered} \min_{\tau, s} \frac{1}{2}(\boldsymbol{\tau}^T Q \boldsymbol{\tau} + \boldsymbol{s}^T R \boldsymbol{s}) \\ B\boldsymbol{\tau} = \boldsymbol{f} + \boldsymbol{s} \\ \boldsymbol{\tau}_{min} \le \boldsymbol{\tau} \le \boldsymbol{\tau}_{max} \end{gathered} \tag{2.6}$$

where $\boldsymbol{s}$ is the vector of slack variables used to penalize $|B\boldsymbol{\tau} - \boldsymbol{f}|$, Q and R are the weight diagonal matrices for the thrusters and DOFs, respectively, $\boldsymbol{\tau}_{min}$ and $\boldsymbol{\tau}_{max}$ are the vector of thrust constraints according to Bollard pull tests.

The thrust allocation of an overactuated AUV with tunnel vertical thrusters is a velocity-dependent problem. For example, the pitch motion of the vehicle on zero velocity is better created by tunnel thrusters due to their large thrust arm. But the effectiveness of a tunnel thruster tends to zero at high velocities (Fig. 2.2).

We propose a variable constraints method for the thrust allocation problem. This method allows reallocating the thrust depending on the vehicle velocity. The method considers thrust constraints that depend on the vehicle velocity, as well as the thrusters' angle of attack:

$$\boldsymbol{\tau}_{min}(v, \theta) \le \boldsymbol{\tau} \le \boldsymbol{\tau}_{max}(v, \theta)$$

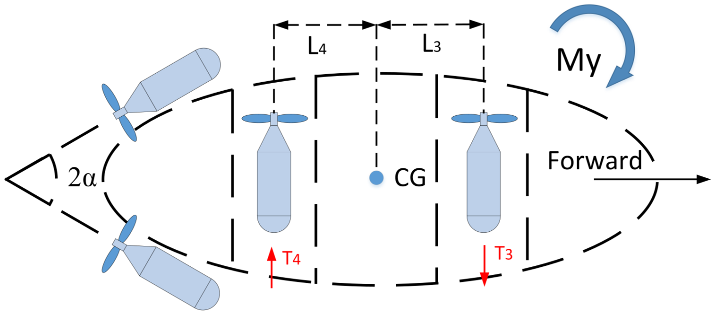
where $\boldsymbol{\tau}_{min}(v, \theta)$ and $\boldsymbol{\tau}_{max}(v, \theta)$ are the variable thrust constraints depending on the vehicle velocity and the thrusters' angle of attack.

3. THE MODEL OF THRUSTER CONSTRAINTS

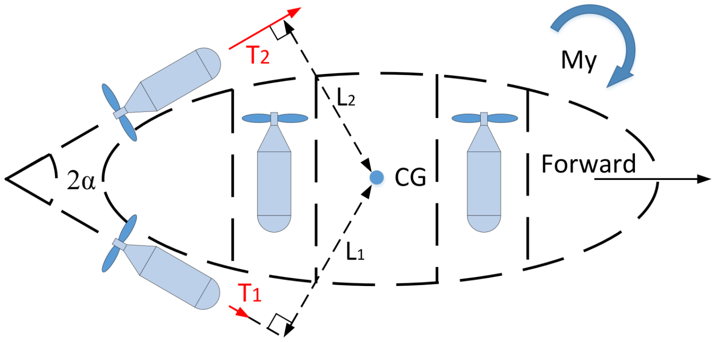
The thrust and torque generated by a thruster can be described by the following formulas:

$$\begin{gathered} \tau = K_\tau(J_0)\rho\Omega\,|\Omega|\,D^4 \\ m = K_m(J_0)\rho\Omega\,|\Omega|\,D^5 \end{gathered}$$

where ρ is the density of water, Ω is the rotational speed of the thruster propeller, and D is the diameter of the propeller. $K_\tau(J_0), K_m(J_0)$ are the coefficients of thrust and torque determined



(a) $v < 0.7ms^{-1}$



(b) $v > 0.7ms^{-1}$

Fig. 2.2. Thrust allocation for pitch motion depending on vehicle velocity.

by the form of the propeller and the characteristics of the motor. $J_0 = v/\Omega D$ is the advance ratio (where v is a velocity of thruster input flow, this velocity equals to velocity of the vehicle in case of the absence of currents).

The function $K_\tau(J_0)$ can be fitted as a linear function of J_0 at $J_0 > 0$ and $v > 0$ [10]:

$$K_\tau^+(J_0) = K_\tau^0(0) - a_1 J_0 \tag{3.7}$$

where $K_\tau^+(J_0)$ is the thrust coefficient $K_\tau(J_0)$ at $J_0 > 0$ and equi-directional state of ambient flow and axial thruster flow. K_τ^0 is the thrust coefficient at $J_0 = 0$ and a_1 is a coefficient determined by the shape of the propeller. The equation may be rewritten as a function of velocity by fixing the value of Ω:

$$K_\tau(v) = K_\tau^0 - b_1 v \tag{3.8}$$

where $b_1 = a_1/D\Omega$.

The thrusters' efficiency also depends on the thrusters' angle of attack θ. At small angle of attack ($\theta < \theta_1^*$, where θ_1^* is the first Critical Incoming Angle [10]), $K_\tau(v, \theta)$ can be written as

$$K_\tau^\theta(v,\theta) = K_\tau^0 + \left(K_\tau^0 - K_\tau^+(v)\right)\left[\sin\left(\frac{\theta}{\theta_1^*}\frac{\pi}{2}\right) - 1\right] \tag{3.9}$$

where $K_\tau^\theta(v,\theta)$ is the thrust coefficient at vehicle velocity equals to v and the thrusters' angle of attack equals to θ, $K_\tau^+(v)$ is determined by equation 3.8, and θ_1^* is the first Critical Incoming Angle: $\theta_1^* = \pi/2 - a_1 v$.

During Bollard pull tests (at $v = 0$), the thrust of the actuators can be obtained as a function of the input code or the current. The maximum thrust of the thruster can be written as a function of K_τ:

$$\tau^{Max} = K_\tau^0 \rho \Omega^{Max} \left|\Omega^{Max}\right| D^4 \tag{3.10}$$

where τ^{Max} is the maximum thrust generated by the thruster during the Bollard pull test and Ω^{Max} is the number of revolution per second of the propeller at maximum thrust.

Hence, the maximum thrust τ^{Max} for a stern thruster at vehicle velocity v and the thrusters' angle of attack θ can be obtained from equations (3.7), (3.9) and (3.10):

$$\tau^{Max}(v,\theta) = \tau^{Max} - Cv\left[1 - \sin\left(\frac{\theta}{\theta_1^*}\frac{\pi}{2}\right)\right] \tag{3.11}$$

where $\tau^{Max}(v,\theta)$ is the maximum thrust generated by the thruster at vehicle velocity v and the thrusters' angle of attack θ, $C = b_1/\rho\Omega^{Max}\left|\Omega^{Max}\right| D^4$. More complicated cases, when $J < 0$ or $v < 0$, can be obtained from the appropriate equations [10].

The steady state performance of a tunnel thruster can be fitted as an exponential function of the velocity ratio v/v_{jet} [11]:

$$\tau(v) = \tau(0)exp\left[-c\left(\frac{v}{v_{jet}}\right)^2\right]$$

where $\tau(v)$ is the the resulting force of the tunnel thruster for a vehicle velocity of v, $\tau(0)$ is the thruster's performance in the Bollard pull test, and v_{jet} is the axial velocity of flow generated by the thruster:

$$v_{jet} = \sqrt{\frac{\tau(0)}{\rho S}}. \tag{3.12}$$

Here, ρ is the density of water and S is the cross sectional area of the tunnel thruster.

Hence, the tunnel thruster constraints can be written as

$$\tau(v)^{max} = \tau(0)^{max}exp\left[-c\left(\frac{v}{v_{jet,lim}}\right)^2\right] \tag{3.13}$$

where $\tau(v)^{max}$ is the maximum thrust when the vehicle's motion has velocity v and $\tau(0)^{max}$ is the thruster constraint provided by the Bollard pull tests of the thruster.

4. PROPULSION SYSTEM SETUP

The model of the AUV "MT-2012" [12] propulsion system was used for algorithm simulation. The propulsion system consists of five thrusters: four thrusters located at the stern of the vehicle at an angle of α to the longitudinal axis (x axis), and a vertical tunnel thruster located at the forward part of the vehicle (Fig. 4.3).

The propulsion system of the vehicle can be described by the thrusters' configuration matrix B [13]:

$$B = [\boldsymbol{b}_b, \boldsymbol{b}_l, \boldsymbol{b}_u, \boldsymbol{b}_r, \boldsymbol{b}_f]$$

where $\boldsymbol{b}_d, \boldsymbol{b}_l, \boldsymbol{b}_u, \boldsymbol{b}_r$ are column vectors for (**d**own, **l**eft, **u**p, **r**ight) the stern thrusters and $\boldsymbol{b}_f$ is the column vector for the **f**orward tunnel thruster. The column vectors in four DOFs (*surge, heave, pitch* and *yaw*) take the form described below.

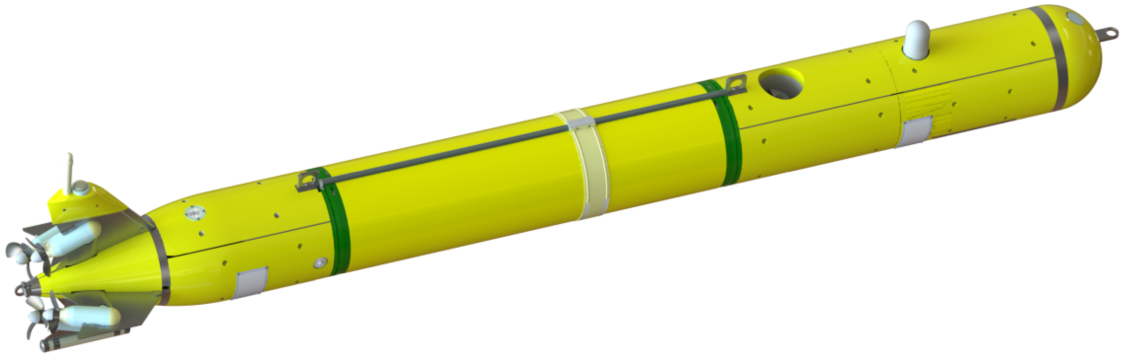


Fig. 4.3. Propulsion system of “MT-2012”.

For the up and down stern thrusters:

$$\boldsymbol{b}_u = \begin{pmatrix} \cos\alpha \\ -\sin\alpha \\ 0 \\ l_z^u \cos\alpha - l_x^u \sin\alpha \end{pmatrix}, \boldsymbol{b}_d = \begin{pmatrix} \cos\alpha \\ \sin\alpha \\ 0 \\ l_z^d \cos\alpha - l_x^d \sin\alpha \end{pmatrix}.$$

For the left and right stern thrusters:

$$\boldsymbol{b}_l = \begin{pmatrix} \cos\alpha \\ 0 \\ l_y^l \cos\alpha - l_x^l \sin\alpha \\ 0 \end{pmatrix}, \boldsymbol{b}_r = \begin{pmatrix} \cos\alpha \\ 0 \\ l_y^r \cos\alpha - l_x^r \sin\alpha \\ 0 \end{pmatrix}.$$

For the vertical tunnel thruster:

$$\boldsymbol{b}_f = \begin{pmatrix} 0 \\ 1 \\ -l_x^{tunnel} \\ 0 \end{pmatrix}$$

where $\boldsymbol{r}^i = [l_x^i, l_y^i, l_z^i]$ is the arm vector of thruster i (where $i = u, d, l, r, f$). For this propulsion system, for reasons of symmetry, $l_z^u = l_y^r = -l_z^d = -l_y^l = L_{zy}^{stern}$ and $l_x^u = l_x^d = l_x^l = l_x^r = L_x^{stern}$.

The parameters of the propulsion system are shown in Table 4.1.

Table 4.1. Parameters of the AUV “MT-2012” propulsion system

L_x^{stern}, m	1.88	$\alpha, grad$	22.5°
L_{zy}^{stern}, m	0.23	f_{lim}^{stern}, N	120.1/-68.2
l_x^{tunnel}, m	1.20	f_{lim}^{tunnel}, N	122.1/-122.0

5. SIMULATION SETUP

The proposed allocation control method with variable constraints was tested on depth maneuvering at different velocities, 0.3, . . . , 1.8 ms^{-1} (corresponding to forward forces 3.6, . . . , 129.6 N, and 50 Nm of pitch torque.

The optimal thrust allocation problem for different velocities was solved by MATLAB Optimization Toolbox. Since $Q > 0$ and $R > 0$ (equation (2.6)) this is a convex quadratic problem in τ, so the interior point method or active set method for convex optimization problem can be used.

The results of the thrust allocation for different vehicle velocities are shown in Fig. 5.4.

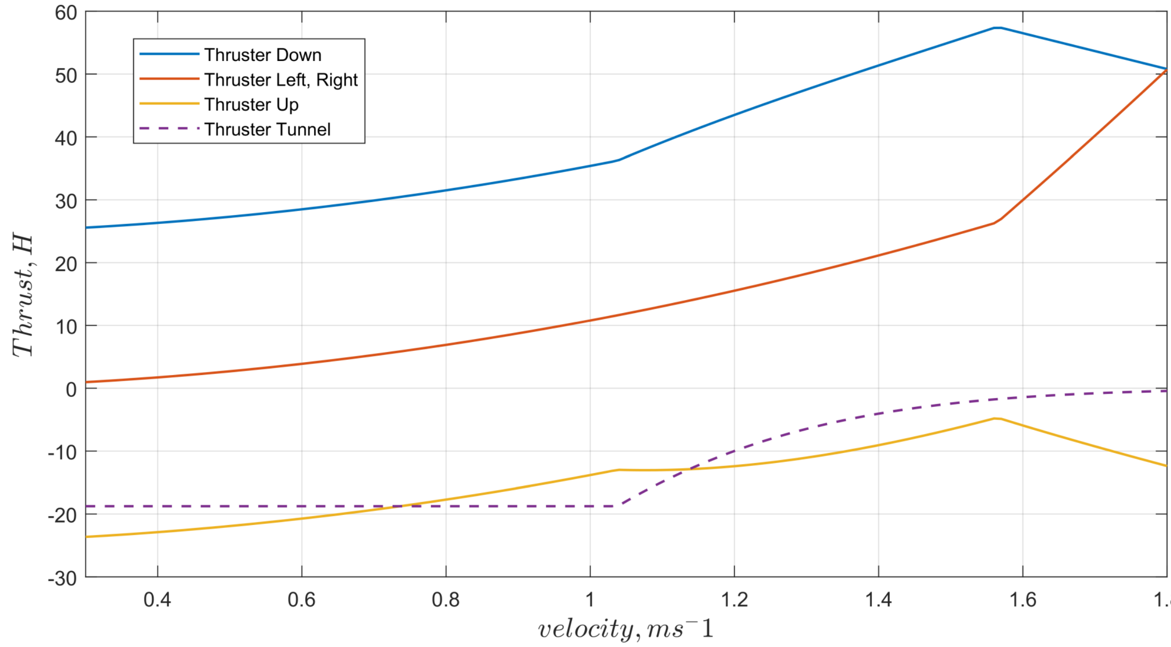


Fig. 5.4. Thrust allocation for different velocities.

There are some implementations of quadratic programming solvers in the C/C++ languages. These implementations are faster than the MATLAB version, and are easy to integrate in vehicle software. There are different types of quadratic programming solvers: qpOASES implements the active set method [14], which allows using a hot start to reduce the calculation time for each new iteration using the data of the previous one, and there are several libraries with implementations of the interior point method: [15] and [16].

6. CONCLUSION

To create a multi-purpose AUV capable of undertaking both survey-style missions and low speed interactions requires an over-actuated propulsion system and a sophisticated thrust allocation algorithm. One method of extending the functionality of the vehicle is installing tunnel thrusters to the vehicle propulsion system.

A new approach for thrust allocation algorithm is proposed in the present article. The method based on the quadratic cost optimal problem with constraints. The main feature of the method is that thrust constraints depend on the vehicle velocity, as well as the thrusters' angle of attack. This method takes into account the decrease in the thrust of tunnel and stern thrusters at high velocities, and reallocates the thrust accordingly.

The proposed algorithm was tested with a simulation model of the real AUV "MT-2012" propulsion system. The propulsion system consists of five thrusters: four thrusters located at the stern of the vehicle and the vertical tunnel thruster located at the forward part of the vehicle. The result shows the adaptive thrust reallocation depending on vehicle velocity in case of depth maneuvering.

REFERENCES

1. Fossen, T. I. & Johansen, T. A. (2013) Control allocation—A survey. *Automatica*, **1**, 1087–1103.
2. Naeem, W., Sutton, R., & Chudley, J. (2006) Soft computing design of a linear quadratic Gaussian controller for an unmanned surface vehicle. In *14th Mediterranean Conference on Control and Automation*, Palermo, Italy, 1–6.
3. Moreira, L. & Guedes Soares, C. (2008) H2 and Hinf designs for diving and course control of an autonomous underwater vehicle in presence of waves. *IEEE Journal of Oceanic Engineering*, **33**, 69–88, https://dx.doi.org/10.1109/ICSMC.2012.6378129.
4. Yan, Z., Chung, S. F., & Wang, J. (2012) Model predictive control of autonomous underwater vehicles based on the simplified dual neural network. In *2012 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, Seoul, Korea, 2551–2556, https://dx.doi.org/10.1109/ICSMC.2012.6378129.
5. Pham, C. D., Spiten, C., & From, P. J. (2015) A control allocation approach to haptic control of underwater robots. In *2015 IEEE International Workshop on Advanced Robotics and its Social Impacts (ARSO)*, Lyon, France, 1–6, https://dx.doi.org/10.1109/ARSO.2015.7428209.
6. Saback, R., Cesar, D., Arnold, S., Lepikson, H., Santos, T., & Albiez, J. (2016) Fault-tolerant control allocation technique based on explicit optimization applied to an autonomous underwater vehicle. In *OCEANS 2016 MTS/IEEE Monterey*, Monterey, CA, 1–8, https://dx.doi.org/10.1109/OCEANS.2016.7761251.
7. Huang, H., Zhang, G., Yang, Y., Xu, J., Li, J., & Wan, L. (2016) Thrust optimal allocation for broad types of underwater vehicles. In *Advances in Swarm Intelligence: 7th International Conference, ICSI 2016*, Bali, Indonesia, 491–502, https://dx.doi.org/10.1007/978-3-319-41009-8_53.
8. Ben, M. & Lee, T. H. (2011) *Unmanned Rotorcraft Systems*. Berlin, Germany: Springer-Verlag.
9. Johansen, T., Fossen, T., & Tøndel, P. (2005) Efficient optimal constrained control allocation via multiparametric programming. *Journal of Guidance, Control, and Dynamics*, **28**, 506–515.
10. Kim, J. (2009) Thruster modeling and controller design for unmanned underwater vehicles (UUVs). In Inzartsev, A. V. (ed.), *Underwater Vehicles*, (pp. 235–250), InTechg, https://dx.doi.org/10.5772/6705.
11. Palmer, A., Hearn, G., & Stevenson, P. (2008) Modelling tunnel thrusters for autonomous underwater vehicles. *IFAC Proceedings Volumes*, **41**, 91–96.
12. Matvienko, Y. V., Boreyko, A. V., Lvov, O. Y., & Vaulin, Y. V. (2015) Kompleks robototehnicheskih sredstv dlja vypolnenija poiskovyh rabot i obsledovanija podvodnoj infrastruktury na shel'fe [Complex Robotic Tools to Perform Searches and Surveys of Underwater Infrastructure on the Shelf]. *Podvodnye issledovaniya i robototekhnika*, **1**, 4–15, [in Russian].
13. Fossen, T. (2011) *Handbook of Marine Craft Hydrodynamics and Motion Control*. Chichester, UK: John Wiley & Sons, Ltd.
14. Ferreau, H., Kirches, C., Potschka, A., Bock, H., & Diehl, M. (2014) qpOASES: A parametric active-set algorithm for quadratic programming. *Mathematical Programming Computation*, **6**, 327–363.
15. Gertz, E. M. & Wright, S. J. (2003) Object-oriented software for quadratic programming. *ACM Transactions on Mathematical Software*, **1**, 58–81.
16. Yan, Y. (2014), C++ implementation of the interior point methods (CPIPM). [Online]. Available https://github.com/YimingYAN/cppipm.

