This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

Design and Evaluation of Dynamic Positioning Controllers With Parasitic Thrust Reduction for an Overactuated Floating Platform

Aristomenis Tsopelakos, *Student Member, IEEE*, and Evangelos Papadopoulos, *Senior Member, IEEE*

***Abstract*— We investigate which control technique is the most suitable for the dynamic positioning of an overactuated platform. To this end, we develop a backstepping and a model predictive controller (MPC). The presence of redundant control inputs makes the stabilization of the position and the orientation of the platform challenging. Settling delays in the actuator thrust and angle, thrust saturation bounds, and jet rotational velocity bounds contribute to the challenge of the problem. To reduce energy consumption, we propose a technique for restricting the parasitic thrust effect. The significant energy reduction due to parasitic thrust restriction is illustrated in tables. The performance of the controllers is demonstrated by simulations, under realistic environmental disturbances, and is compared with that of a model-based PID controller previously developed, while the platform accomplishes two typical tasks. The evaluation criteria include energy consumption, robustness, and accuracy of the dynamic positioning. Results show the superiority of the MPC.**

***Index Terms*— Backstepping (BS), control systems evaluation, dynamic positioning, model predictive controller (MPC), overactuated robotics.**

I. Introduction

SEA platforms are used in a wide range of applications in the industrial and scientific sector. The petroleum industry has instated sea platforms to particular sites for the extraction and processing of oil and natural gas. Sea platforms are also used for the submergence of equipment to be assembled under water. Fixed-platform costs are high because they are designed for long term use and they need continuous maintenance because of salt water erosion. To reduce costs, moving floating platforms are introduced. Autonomous floating platforms must have the capability for dynamic positioning. To this end, motion control and dynamic positioning techniques have been proposed in [1] and [2]. A control allocation scheme must be applied on overactuated platforms for the resolution of the redundant control inputs. A survey on control allocation is presented in [3].

Floating platforms are overactuated with maneuver capabilities. However, the redundant control inputs, the settling delays in the actuator thrust and angle, the hardware limitations, the strong environmental disturbances, and the need for minimum energy consumption render the problem of their dynamic positioning challenging.

Fig. 1. Vereniki.

Methodologies have been presented for the robust dynamic positioning of marine vessels [4], for the tracking control of overactuated surface vessels [5], and for the heading control of yachts [6] and ships [7]. Results on the compensation of sideslips forces acting on marine vessels are presented in [8]. The development of an adaptive controller for ships with partially unknown dynamics is presented in [9]. In these works, the authors focus on the mathematical formulation of the controller without taking into account settling delays and hardware limitations inserted by the actuation mechanisms and actuator dynamics.

Backstepping (BS) and model predictive controller (MPC) [25] are widely applied control techniques for marine vessels such as ships [10], [11], underwater vessels [12], [13], and surface vessels [14], [15]. However, very few papers present studies about the performance of the developed controllers. A study on controller performance of marine robots is presented in [16]. Comparative studies are even more rare.

The overactuated floating platform, named Vereniki (see Figs. 1 and 2), was designed to deploy a deep-sea telescope for the detection of neutrinos [17]. The initial study on the modeling and control of Vereniki can be found in [18]. A detailed description of the model, the allocation scheme, the environmental forces, namely, wind, wave, and current forces, and the design of the model-based PID (MB-PID) controller are presented in [19] and [22]. The design of a BS controller for Vereniki and the proposal of a heuristic for energy reduction were presented in [20]. An initial design of the MPC for Vereniki was presented in [21]. In the design of

Manuscript received January 27, 2016; accepted April 3, 2016. Manuscript received in final form April 5, 2016. Recommended by Associate Editor G. Rovithakis. *(Corresponding author: Aristomenis Tsopelakos.)*

A. Tsopelakos is with the Department of Electrical and Computer Engineering, University of Illinois at Urbana–Champaign, Urbana, IL 61801 USA (e-mail: tsopela2@illinois.edu).

E. Papadopoulos is with the School of Mechanical Engineering, National Technical University of Athens, Athens 15780, Greece (e-mail: egpapado@central.ntua.gr).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCST.2016.2554060

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

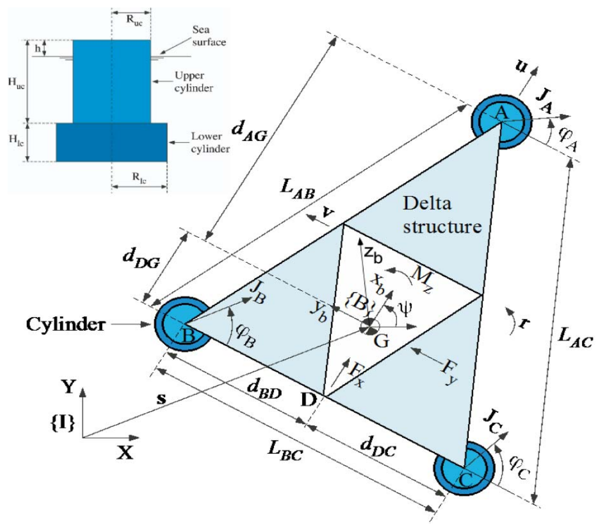

Fig. 2. Vereniki geometrical representation.

the controllers of Vereniki, settling delays in the actuator thrust and angle, thrust saturation bounds, jet rotational velocity bounds, and actuator dynamics are taken into consideration.

We improve the BS and the MPC, we evaluate them, and we compare them with an MB-PID developed previously [19], [22]. In Section II, we describe the mechanical characteristics of the platform, the hydrodynamic and environmental forces, the kinematics and dynamics, and the allocation scheme.

In Section III, we present the problem of parasitic thrust and we propose a novel technique for its restriction. The restriction of parasitic thrust will considerably reduce the energy consumed. The energy reduction for each controller is presented in Tables VI and XVI in Section V. The proposed technique can be applied to all vectored thrust vessels, surface, underwater, or aerial, with considerable reduction in the energy consumption.

In Section IV, we present the formulation of the controllers. BS is given in detail, while in [20], we just give the formula of the controls. BS accurately handles settling delays in the development of the desired forces and torque on the center mass (CM) of the platform produced by the actuators. These delays are considered as some kind of disturbance in other controllers, without any handling.

Compared with [21], the MPC is improved. In the previous MPC, for a large prediction horizon, the optimization problem was becoming ill-conditioned, meaning that slight changes in the state vector could result in considerable changes on the control input. This is unwanted when we have saturation limits. In this paper, MPC handles this issue.

The evaluation presented in Section V was based on three criteria: 1) robustness; 2) energy consumption; and 3) accuracy of positioning, while the platform accomplishes two tasks. The robustness in mass is an important feature, as these platforms carry loads. The introduced performance index (PI) correlates energy consumption and accuracy positioning. Based on simulations, we make a comparative study, which concludes with the superiority of the MPC in both tasks.

II. Description and Modeling of the Platform Vereniki

Vereniki is an isosceles platform with a double cylinder at each corner. The hollow cylinders contain the jet diesel engines and the electrohydraulic motors responsible for the rotation of the jets that apply thrust parallel to the sea surface. This configuration restricts the motion of the platform on the seaplane. The platform position and orientation are obtained through GPS sensors [19]. In order to filter the noise inserted from the measurements, the wind, and the waves, we use a Butterworth filter. The controller compensates low-frequency noise interference and the low-pass Butterworth cuts the uncompensated high-frequency interference [22]. Hardware limitations include the thrust upper limit of 20 kN and the jet angular velocity limit of 0.84 rad/s. Thrust is never set to zero. The settling time in developing the jet *thrust* response is about 8 s and the jet *angle* response is about 4 s. The jet thrust J and angle rotation φ dynamics are approximated by a first-order system based on data from the manufacturer of the jets

$$\begin{aligned} \dot{J}_i &= (1/\tau_J)(J_{i,\text{des}} - J_i) \\ \dot{\varphi}_i &= (1/\tau_\varphi)(\varphi_{i,\text{des}} - \varphi_i) \end{aligned} \tag{1}$$

where τ_J and τ_φ are the jet thrust and the rotation time constant, for $i = A, B, C$. $J_i, \varphi_i \in R$. In simulation, $\tau_J = 2$ s and $\tau_\varphi = 1$ s. The desired thrust $J_{i,\text{des}}$ is the thrust we would like to be produced by each jet, respectively. The desired angle $\varphi_{i,\text{des}}$ is the direction we would like each jet to have.

A. Hydrodynamic Forces and Environmental Forces

The hydrodynamic force acting on each cylinder includes two terms. The first term is the added mass force, which is a linear function of the acceleration of each cylinder. The second term is the drag force, which is a quadratic function of the velocity of each cylinder (see [23], [24]). The normal to the axis of each cylinder force on the double-cylinder structure at point A (see Fig. 2) expressed in body-fixed frame $\{B\}$ is given according to Morison's equation by

$$\begin{aligned} {}^B\mathbf{f}_{h,A} &= C_a \pi \rho_w \big[R_{\text{uc}}^2 (H_{\text{uc}} - h) + R_{\text{lc}}^2 H_{\text{lc}}\big]\big({}^B\dot{v}_{\text{wat}}^{\perp} - {}^B\mathbf{a}_A\big) \\ &\quad + C_d \rho_w [R_{\text{uc}}(H_{\text{uc}} - h) + R_{\text{lc}} H_{\text{lc}}] \big\| \big({}^B\mathbf{v}_{\text{wat}}^{\perp} - {}^B\mathbf{v}_A\big) \big\| \\ &\quad \cdot \big({}^B\mathbf{v}_{\text{wat}}^{\perp} - {}^B\mathbf{v}_A\big) \end{aligned} \tag{2}$$

where ρ_w is the water density, C_a is the added mass coefficient, and C_d the drag coefficient. ${}^B v_A$ and ${}^B\mathbf{a}_A$ are the velocity and acceleration of cylinder A, respectively. ${}^B v_{\text{wat}}^{\perp}$ is the component of the velocity vector of the water normal to the axis of each cylinder, calculated from the vectorial addition of the sea current velocity, and the water flow velocity due to sea waves, all expressed in the body-fixed frame [19], [22]. Parameters h, R_{uc}, H_{uc}, R_{lc}, and H_{lc} denote the height of the cylinder above the water surface and the radius and height of the upper and lower cylinder sections, respectively (see [19], [22] and Fig. 2).

The hydrodynamic forces on A given by (2) result in a force acting on the platform CM: ${}^B\mathbf{f}_{h,A}^T$ and a moment about it: $({}^B\mathbf{s}_{A/G} \times {}^B\mathbf{f}_{h,A})$, where ${}^B\mathbf{s}_{A/G}$ is the position of point A

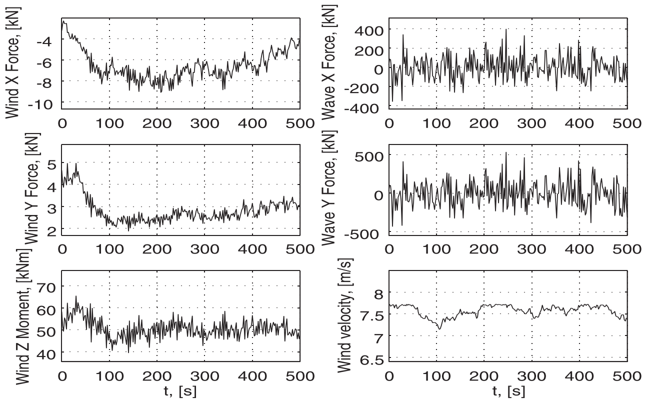

Fig. 3. Environmental disturbances based on data from [17].

with respect to G expressed in {B} (see [19], [22] and Fig. 2). All terms of the force ${}^{B}\mathbf{f}_{h,A}^{T}$ and the torque $({}^{B}\mathbf{s}_{A/G} \times {}^{B}\mathbf{f}_{h,A})$ that are quadratic functions of the velocity of the platform are collected in vector

$$\mathbf{q} = [f_x, f_y, m_z]^T \tag{3}$$

where f_x and f_y are hydrodynamic forces along the x- and y-axis, respectively, and m_z is the torque along the z-axis. The terms that depend linearly on water speed and acceleration are incorporated in the models that generate the wave forces that act on the platform (see [2]). Simulation models also generate the wind and current forces [2]. Wind, wave, and current generated disturbance forces and torques are gathered in $\mathbf{q}_{\text{dist}}$. Both $\mathbf{q}, \mathbf{q}_{\text{dist}} \in R^3$. The controllers have no information about the environmental disturbances. We have to tune the controllers in a way such that the platform converges to (0, 0) despite the disturbances $\mathbf{q}_{\text{dist}}$. Some characteristic values used in the simulation are: $d_{\text{AG}} = 26.694$ m, $d_{\text{BD}} = 18.241$ m, $R_{\text{uc}} = 2.2$ m, $H_{\text{uc}} = 6.3$ m, $R_{\text{lc}} = 3.4$ m, $H_{\text{lc}} = 3.0$ m, $C_a = 0.55$, $C_d = 0.8$, and $\rho = 1025$ kg/m^3.

The hydrodynamic forces/torque acting on the platform CM and the simulation models of forces/torque due to wind, waves, and currents are described in detail in [2], [19], and [22]. Fig. 3 shows a typical sample of wind and wave forces used in the simulation. Inertial sea current speed is $v_c(t) \leq 0.514$ m/s (1 kn) and wind speed is $v_w(t) \leq 7.9$ m/s (15 kn). Maximum values are retrieved from meteorological data [17]. Since the platform will perform on days with low wave height of Douglas scale 0–2, we do not take into account the forces from the waves on the z-axis of the platform. On the simulations, the direction of the wind and wave forces is against the motion of the platform for maximum resistance. In this way, we test the platform in extreme environmental conditions.

B. Kinematics and Dynamics

The kinematic equation for the platform planar motion is

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} c\psi & -s\psi & 0 \\ s\psi & c\psi & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} u \\ v \\ r \end{bmatrix} \Rightarrow \dot{\mathbf{x}} = R\mathbf{v} \tag{4}$$

where $\mathbf{x} = [x, y, \psi] \in R^3$, $\mathbf{v} = [u, v, r] \in R^3$, and $\mathbf{R}_{3\times3}$ is the rotation matrix. The variables x and y are platform CM inertial coordinates defined at the inertial frame $\{I\}$ (see Fig. 2). In Fig. 2, point G stands for the CM of the platform. Variable ψ represents the orientation of the body-fixed frame $\{B\}$ with origin at the platform CM (see Fig. 2). Surge, sway, and yaw (angular) velocities are represented by u, v, and r, respectively, defined in the body-fixed frame $\{B\}$ (see Fig. 2); $s\cdot = \sin(\cdot)$; $c\cdot = \cos(\cdot)$. Assuming that the CM of the platform is at the triangle centroid and that the motion is strictly planar, the equation of motion on $\{B\}$ is

$$\mathbf{M}\dot{\mathbf{v}} = \mathbf{q} + \mathbf{q}_{\text{dist}} + \boldsymbol{\tau}_c \tag{5}$$

where $\mathbf{q}$ is given in (3), $\mathbf{q}_{\text{dist}}$ represents wind, wave, and current generated disturbance forces and torques, and $\boldsymbol{\tau}_c = [F_x, F_y, M_z]^T \in R^3$ represents the forces/torque developed on the CM due to the vectored thrusts provided by the jets. The mass and added mass matrix $\mathbf{M}_{3\times3}$ is

$$\begin{aligned} \mathbf{M} &= \text{diag}(m - 3m_a, m - 3m_a, m_{33}) \\ m_{33} &= I_{zz} - \left(d_{\text{AG}}^2 + 2d_{\text{BD}}^2 + 2d_{\text{DG}}^2\right)m_a \\ m_a &= -\rho\pi C_a\left(R_{\text{uc}}^2(H_{\text{uc}} - h) + R_{\text{lc}}^2 H_{\text{lc}}\right) \end{aligned} \tag{6}$$

where m is the mass of the platform, m_a is the added mass, and I_{zz} is the mass moment of inertia about the z_b-axis (see [19], [22]). The axes x_b, y_b, and z_b form an orthogonal system fixed on the platform CM (G) (see Fig. 2). The values used in the simulation are: $m = 425 * 10^3$ kg, $m_a = -7.791 * 10^4$ kg, and $I_{zz} = 1.8159 * 10^8$ kgm^2. The kinematics, dynamics, matrix $\mathbf{M}$, and added mass m_a are described in detail in [19] and [22].

C. Control Allocation

The controllers compute the desired forces ($F_{x,\text{des}}$, $F_{y,\text{des}}$) and desired torque $M_{z,\text{des}}$ to be applied to the CM of the platform. The desired forces and torque must be resolved to desired jet thrusts ($J_{A,\text{des}}$, $J_{B,\text{des}}$, $J_{C,\text{des}}$) and desired jet angles ($\varphi_{A,\text{des}}$, $\varphi_{B,\text{des}}$, $\varphi_{C,\text{des}}$), which denote the directions of jets vectored thrust. A control allocation scheme is required for the resolution of desired forces and torque to desired thrusts and desired angles (see Fig. 7 for BS and Fig. 8 for MPC). Forces and torque, applied to the platform CM, are related to jet thrusts and angles, according to the following relation:

$$\boldsymbol{\tau}_{c,\text{des}(3\times1)} = [F_{x,\text{des}}, F_{y,\text{des}}, M_{z,\text{des}}]^T = \mathbf{B}_{3\times6}\mathbf{f}_{c,\text{des}(6\times1)} \tag{7}$$

$$\mathbf{B} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & -1 & -d_{\text{AG}} \\ 1 & 0 & -d_{\text{DC}} \\ 0 & -1 & d_{\text{DG}} \\ 1 & 0 & d_{\text{DC}} \\ 0 & -1 & d_{\text{DG}} \end{bmatrix}^T, \quad \mathbf{f}_{c,\text{des}} = \begin{bmatrix} J_{A,\text{des}} s\varphi_{\text{A,des}} \\ J_{A,\text{des}} c\varphi_{\text{A,des}} \\ J_{B,\text{des}} s\varphi_{B,\text{des}} \\ J_{B,\text{des}} c\varphi_{B,\text{des}} \\ J_{C,\text{des}} s\varphi_{C,\text{des}} \\ J_{C.des} c\varphi_{C,\text{des}} \end{bmatrix}. \tag{8}$$

The dimensional parameters in $\mathbf{B}$ are defined in Fig. 2. Two steps define the control allocation scheme.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

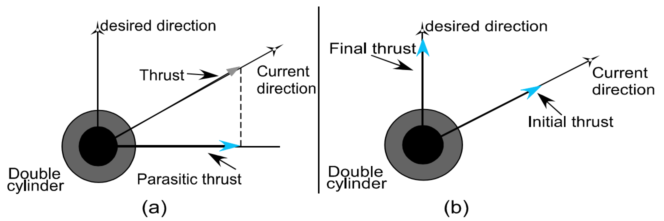

Fig. 4. (a) Parasitic thrust. (b) Ideal response.

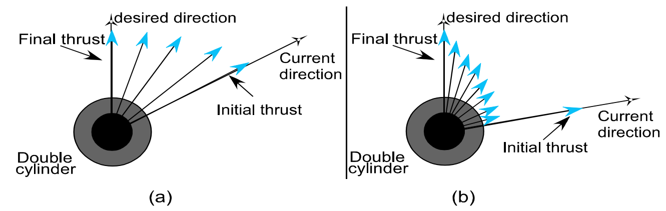

Fig. 5. (a) Real response. (b) Thrust manipulation.

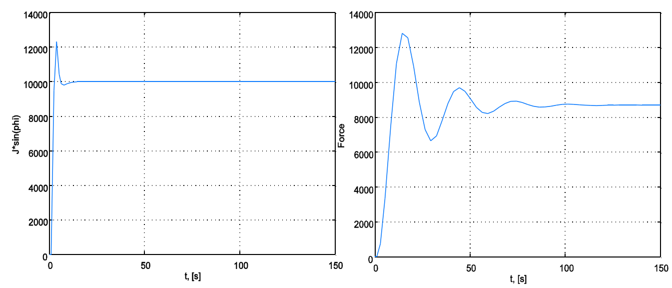

Fig. 6. $F = J * \sin(\varphi)$; J, φ separately (left); F (right).

1) Pseudoinversion of $\mathbf{B}$, yielding $\mathbf{f}_{c,\text{des}}$.
2) Using $\mathbf{f}_{c,\text{des}}$ that contains the components of desired thrusts along the x- and y-axis $\{I\}$, we get desired thrusts and angles

$$\begin{aligned} J_{i,\text{des}} &= \sqrt{(J_{i,\text{des}}\, s\varphi_{i,\text{des}})^2 + (J_{i,\text{des}}\, c\varphi_{i,\text{des}})^2} \\ \varphi_{i,\text{des}} &= a\tan 2(J_{i,\text{des}}\, s\varphi_{i,\text{des}}, J_{i,\text{des}}\, c\varphi_{i,\text{des}}), \\ &\qquad 0 \leq \varphi_{i,\text{des}} < 2\pi \end{aligned} \tag{9}$$

where $i = A, B, C$.

The signals of the desired thrusts and the desired angles will be driven to the actual actuators of the platform, which are the rotating jets (see Figs. 7 and 8). The thrust and the angle response of the rotating jets are given by (1). As a result, the jets will produce the forces (F_x, F_y) and the torque M_z (7) that will drive the platform to a given position and with a given orientation.

Since the minimization of the energy consumed is the most important criterion of the evaluation process, we have to take advantage of the overactuated configuration. The pseudoinversion of $\mathbf{B}$ in (8) results the vector $\mathbf{f}_{c,\text{des}}$ with the minimum norm between all the possible solutions of (7)

$$\min(\| f_{c,\text{des}} \|) = \min\left(\sqrt{J_A^2 + J_B^2 + J_C^2}\right) \Rightarrow \min E. \tag{10}$$

The fix angle configuration for our platform presented in [18] consumes more energy and restricts the dynamic positioning capabilities of the platform.

III. Parasitic Thrust Phenomenon

At the end of each computational cycle, the controller sends to the actuation systems a set of desired thrusts and angles. Hardware limitations, settling delays, and environmental disturbances prevent the jets from reaching the desired angles instantly. The vectored thrust attains the desired direction after several control loops. The thrust component normal to the desired direction is a parasitic thrust [see Fig. 4(a)]. The ideal response is depicted in Fig. 4(b).

The parasitic thrust is a common phenomenon to systems controlled by vectored thrust, and in most cases, it is considered as a disturbance, without any special treatment. In this paper, we propose a technique to restrict the effect of the parasitic thrust. The main idea is that it is preferable to have very low thrust when this is too far from the desired direction. So, $J_{i,\text{des}}$ is multiplied with the factor $c_i \in R$

$$c_i = \exp(-a|\varphi_{i,\text{des}} - \varphi_i|). \tag{11}$$

The coefficient $a \in R$ is selected so that when $|\varphi_{i,\text{des}} - \varphi_i|$ is close to 180°, c_i is close to zero. With this technique, we reduce the energy consumed for the counteraction of the parasitic thrust. Fig. 5(a) shows the real response of the jet. Fig. 5(b) illustrates the use of (11). The thrust is reduced during jet rotation to be restored to the required thrust when $\varphi_{i,\text{des}} = \varphi_i$. In the simulation, $a = 0.735$.

IV. Design of Controllers

A. Backstepping Controller

In BS, we accurately handle the settling delays. Since there exists a settling delay in the development of the desired thrusts and angles, we assume that there must also exist a settling delay in the development of the desired forces and torque on platform CM. Jet thrust and angle response are modeled by a first-order system (1). The development of the forces and the torque on platform CM is modeled by a second-order system, as it can capture overshoots in the response generated by multiplications of the thrust J_i with the $s(\cdot)$ or $c(\cdot)$ of the angles, which does not a first-order system (see Fig. 6). So, we have

$$\ddot{\boldsymbol{\tau}}_c = -(1/\tau_1)\dot{\boldsymbol{\tau}}_c - (1/\tau_2)\boldsymbol{\tau}_c + (1/\tau_3)\boldsymbol{\tau}_{c,\text{des}} \tag{12}$$

where τ_1, τ_2, and τ_3 are time constants and $\tau_{c,\text{des}}$ is the input vector

$$\boldsymbol{\tau}_{c,\text{des}} = [F_{x,\text{des}}, F_{y,\text{des}}, M_{z,\text{des}}]^T = [u_{F_x}, u_{F_y}, u_{M_z}]^T. \tag{13}$$

The desired forces/torque are the controller inputs. In Fig. 6, we can see the response of $F = J \cdot \sin(\varphi)$ in two cases. On the left, we order J to change from 10 N to 20 kN and φ from 0° to 150°, J and φ being governed by (1); and we plot the result of the multiplication. On the right, we order quantity F to change from 10 to $20000 * \sin(150)$, F is governed by (12), with $\tau_1 = 10$ s, $\tau_2 = 20$ s, and $\tau_3 = 23$ s. We see that (12) captures the overshoot.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

We cannot apply the representation (12) in the MB-PID or MPC because in the design of these controllers the forces/torque are used as control variables for the cancellation of the nonlinearities to yield a linear system with bounded disturbance [see (79)]. For BS, the system under control is (4), (5), and (12). Term $\mathbf{q}_{\text{dist}}$ is treated as an external disturbance. In order to make it easier for the reader to follow, the system is repeated in the following equations:

$$\dot{\mathbf{x}} = \mathbf{R}\mathbf{v} \tag{14}$$

$$\dot{\mathbf{v}} = \mathbf{M}^{-1}(\mathbf{q} + \boldsymbol{\tau}_c) \tag{15}$$

$$\ddot{\boldsymbol{\tau}}_c = -(1/\tau_1)\dot{\boldsymbol{\tau}}_c - (1/\tau_1)\boldsymbol{\tau}_c + (1/\tau_1)\boldsymbol{\tau}_{c,\text{des}}. \tag{16}$$

We consider the following transformation for the second-order system (16):

$$\begin{aligned} \dot{\boldsymbol{\tau}}_c &= \mathbf{p} \\ \dot{\mathbf{p}} &= -(1/\tau_1)\mathbf{p} - (1/\tau_1)\boldsymbol{\tau}_c - (1/\tau_1)\boldsymbol{\tau}_{c,\text{des}} \end{aligned} \tag{17}$$

where

$$\mathbf{p} = [p_x, p_y, p_z]^T = [\dot{F}_x, \dot{F}_y, \dot{M}_z]^T. \tag{18}$$

The transformation described in (17) and (18) for a second-order system can be inductively extended to higher order systems. The BS controller we develop does not depend on the order of the dynamics which describe the development of the forces and the torque on platform CM. The BS design with first-order system in forces/torque was given in [20].

1) Preliminary Computations: The reference position, direction, and velocities are denoted by x_R, y_R, ψ_R, u_R, v_R, and r_R, respectively. The tracking errors are defined as $x_e = x - x_R$, $y_e = y - y_R$, $\psi_e = \psi - \psi_R$, $u_e = u - u_R$, $v_e = v - v_R$, and $r_e = r - r_R$. After the substitution of the tracking errors into (4), the following equations result:

$$\begin{bmatrix} \dot{x}_e \\ \dot{y}_e \end{bmatrix} = \begin{bmatrix} c\psi & -s\psi \\ s\psi & c\psi \end{bmatrix} \begin{bmatrix} u_e \\ v_e \end{bmatrix} + \begin{bmatrix} \delta_1 \\ \delta_2 \end{bmatrix} \tag{19}$$

$$\dot{\psi}_e = r_e \tag{20}$$

$$\boldsymbol{\delta} = \begin{bmatrix} \delta_1 \\ \delta_2 \end{bmatrix} = \begin{bmatrix} c\psi - c\psi_R & -s\psi + s\psi_R \\ s\psi - s\psi_R & c\psi - c\psi_R \end{bmatrix} \begin{bmatrix} u_R \\ v_R \end{bmatrix}. \tag{21}$$

Since u_R and v_R are bounded desired velocities, and quantities δ_1 and δ_2 are treated as bounded disturbance. For a stabilized ψ ($\psi \to \psi_R$), this disturbance tends to zero.

2) Stabilization Process:

a) Step 1: We begin with the stabilization of (19). The linear velocities u_e and v_e are the virtual control variables. The desired values for the virtual controls u_e and v_e are given by

$$\begin{aligned} \begin{bmatrix} u_{e,\text{des}} \\ v_{e,des} \end{bmatrix} &= -\begin{bmatrix} c\psi & s\psi \\ -s\psi & c\psi \end{bmatrix} [\mathbf{K} + \mathbf{K}_1] \begin{bmatrix} x_e \\ y_e \end{bmatrix} \\ \mathbf{K} = \text{diag}(k, k), &\quad \mathbf{K}_1 = \text{diag}(k_1, k_1). \end{aligned} \tag{22}$$

The reason for choosing these desired values is that their substitution into (19) results in an exponentially decreasing response augmented by bounded disturbances, reassuring convergence to (0, 0)

$$\begin{bmatrix} \dot{x}_e \\ \dot{y}_e \end{bmatrix} = -\begin{bmatrix} k + k_1 & 0 \\ 0 & k + k_1 \end{bmatrix} \begin{bmatrix} x_e \\ y_e \end{bmatrix} + \begin{bmatrix} \delta_1 \\ \delta_2 \end{bmatrix} \tag{23}$$

where k and k_1 are positive numbers.

Using (19) and (22), we compute the derivatives of $u_{e,\text{des}}$ and $v_{e,\text{des}}$. To simplify the expressions, we take advantage of the fact that the angular rate of the platform and as such of the reference frame is so low that it can be neglected

$$\begin{aligned} \dot{u}_{e,\text{des}} &= -(k + k_1)u_e - (k + k_1)(\delta_1 c\psi + \delta_2 s\psi) \\ \dot{v}_{e,\text{des}} &= -(k + k_1)v_e - (k + k_1)(-\delta_1 s\psi + \delta_2 c\psi). \end{aligned} \tag{24}$$

We define the following Lyapunov function, as if we had to stabilize only subsystem (19):

$$V_1 = (1/2)x_e^2 + (1/2)y_e^2 \tag{25}$$

$$\begin{aligned} \dot{V}_1 = &-kx_e^2 - ky_e^2 - k_1\left(x_e - \frac{\delta_1}{2k_1}\right)^2 \\ &- k_1\left(y_e - \frac{\delta_2}{2k_1}\right)^2 + \frac{\|\boldsymbol{\delta}\|^2}{4k_1}. \end{aligned} \tag{26}$$

Both k and k_1 gains are necessary to formulate the quadratic form (26). All terms in $\dot{V}_1$ are negative except to $||\boldsymbol{\delta}||^2/4k_1$ which can be very small, by choosing a large k_1. The linear velocities u_e and v_e are virtual controls. We cannot attribute to them desired values, but we can make them converge to these desired values. To this end, we introduce the following error variables and stabilize these to (0, 0):

$$\begin{aligned} z_u &= u_e - u_{e,\text{des}} \Rightarrow u_e = z_u + u_{e,\text{des}} \\ z_v &= v_e - v_{e,\text{des}} \Rightarrow v_e = z_v + v_{e,\text{des}}. \end{aligned} \tag{27}$$

Since we have defined these two new variables, we have to make a change of variables in (19). Substituting (22) and (27) into (19) yields the following subsystem:

$$\begin{aligned} \begin{bmatrix} \dot{x}_e \\ \dot{y}_e \end{bmatrix} = &-\begin{bmatrix} k + k_1 & 0 \\ 0 & k + k_1 \end{bmatrix} \begin{bmatrix} x_e \\ y_e \end{bmatrix} \\ &+ \begin{bmatrix} c\psi & -s\psi \\ s\psi & c\psi \end{bmatrix} \begin{bmatrix} z_u \\ z_v \end{bmatrix} + \begin{bmatrix} \delta_1 \\ \delta_2 \end{bmatrix}. \end{aligned} \tag{28}$$

Based on (15), the derivatives of the linear velocities are

$$\begin{aligned} \dot{u}_e &= (1/(m - 3m_a))(f_x + F_x) - \dot{u}_R \\ \dot{v}_e &= (1/(m - 3m_a))(f_y + F_y) - \dot{v}_R. \end{aligned} \tag{29}$$

To drive z_u and z_v to (0, 0), we have to describe their dynamics. Using (24), (27), and (29), we compute the error derivatives

$$\begin{aligned} \dot{z}_u = &(1/(m - 3m_a))f_x - \dot{u}_R + (k + k_1)u_e \\ &+ (k + k_1)(\delta_1 c\psi + \delta_2 s\psi) + (1/(m - 3m_a))F_x \\ \dot{z}_v = &(1/(m - 3m_a))f_y - \dot{v}_R + (k + k_1)v_e \\ &- (k + k_1)(\delta_1 s\psi - \delta_2 c\psi) + (1/(m - 3m_a))F_y. \end{aligned} \tag{30}$$

b) Step 2: We continue with the stabilization of system (30). The variables F_x and F_y are the virtual controls. To stabilize (30), we compute the desired forces $F_{x,\text{des}}$ and $F_{y,\text{des}}$ and make the forces F_x and F_y converge to them. We update the Lyapunov

$$V_2 = V_1 + (1/2)\left(z_u^2 + z_v^2\right). \tag{31}$$

Then, we compute its derivative

$$\begin{aligned}\dot{V}_2 = {} & \dot{V}_1 + z_u((1/(m-3m_a))f_x - \dot{u}_R + (k+k_1)u_e \\ & + (k+k_1)(\delta_1 c\psi + \delta_2 s\psi) + (x_e c\psi + y_e s\psi) \\ & + (1/(m-3m_a))F_x) + z_v((1/(m-3m_a))f_y - \dot{v}_R \\ & + (k+k_1)v_e - (k+k_1)(\delta_1 s\psi - \delta_2 c\psi) \\ & + (-x_e s\psi + y_e c\psi) + (1/(m-3m_a))F_y). \qquad (32)\end{aligned}$$

After the introduction of z_u and z_v, the derivatives of x_e and y_e are given by (28). The desired forces (virtual controls) are

$$\begin{aligned}F_{x,\text{des}} = {} & (m-3m_a)(-(1/(m-3m_a))f_x + \dot{u}_R \\ & - (k+k_1)u_e - (k+k_1)(\delta_1 c\psi + \delta_2 s\psi) \\ & - (x_e c\psi + y_e s\psi) - c_u z_u) \\ F_{y,\text{des}} = {} & (m-3m_a)(-(1/(m-3m_a))f_y + \dot{v}_R \\ & - (k+k_1)v_e + (k+k_1)(\delta_1 s\psi - \delta_2 c\psi) \\ & - (-x_e s\psi + y_e c\psi) - c_v z_v). \qquad (33)\end{aligned}$$

The Lyapunov derivative takes the following form:

$$\dot{V}_2 = \dot{V}_1 - c_u z_u^2 - c_v z_v^2. \qquad (34)$$

The variables F_x and F_y are the virtual controls. So, for the same reason as in (27), we define the following errors:

$$\begin{aligned}z_{F_x} &= F_x - F_{x,\text{des}} \Rightarrow F_x = z_{F_x} + F_{x,\text{des}} \\ z_{F_y} &= F_y - F_{y,\text{des}} \Rightarrow F_y = z_{F_y} + F_{y,\text{des}}. \qquad (35)\end{aligned}$$

Based on (18), we compute the dynamics of the errors (35)

$$\begin{aligned}\dot{z}_{F_x} &= p_x - \dot{F}_{x,\text{des}} \\ \dot{z}_{F_y} &= p_y - \dot{F}_{y,\text{des}}. \qquad (36)\end{aligned}$$

Since we have defined two new variables, we have to make the change of variables in (30). After substitution of (33) and (35) into (30), we obtain the following subsystem:

$$\begin{aligned}\dot{z}_u &= -c_u z_u - (x_e c\psi + y_e s\psi) + (1/(m-3m_a))z_{F_x} \\ \dot{z}_v &= -c_v z_v - (-x_e s\psi + y_e c\psi) + (1/(m-3m_a))z_{F_y}. \qquad (37)\end{aligned}$$

c) Step 3: We will stabilize subsystem (36). The updated Lyapunov function has the following form:

$$V_3 = V_2 + \frac{1}{2}z_{F_x}^2 + \frac{1}{2}z_{F_y}^2. \qquad (38)$$

After derivation of the Lyapunov function, we compute $p_{x,\text{des}}$ and $p_{y,\text{des}}$ [see (18)], so that all terms are negative except to $||\boldsymbol{\delta}||^2/4k_1$

$$\begin{aligned}p_{x,\text{des}} &= \dot{F}_{x,\text{des}} - (1/(m-3m_a))z_u - c_{F_x} z_{F_x} \\ p_{y,\text{des}} &= \dot{F}_{y,\text{des}} - (1/(m-3m_a))z_v - c_{F_y} z_{F_y}. \qquad (39)\end{aligned}$$

The derivative of V_3 becomes

$$\dot{V}_3 = \dot{V}_2 - c_{F_x} z_{F_x}^2 - c_{F_y} z_{F_y}^2. \qquad (40)$$

We also compute the respective errors and their time derivatives, according to (17)

$$\begin{aligned}z_{p_x} &= p_x - p_{x,\text{des}} \Rightarrow p_x = z_{p_x} + p_{x,\text{des}} \\ z_{p_y} &= p_y - p_{y,\text{des}} \Rightarrow p_y = z_{p_y} + p_{y,\text{des}} \qquad (41) \\ \dot{z}_{p_x} &= -(1/\tau_1)p_x - (1/\tau_1)F_x - \dot{p}_{x,\text{des}} + (1/\tau_1)u_{F_x} \\ \dot{z}_{p_y} &= -(1/\tau_1)p_y - (1/\tau_1)F_y - \dot{p}_{y,\text{des}} + (1/\tau_1)u_{F_y}. \qquad (42)\end{aligned}$$

Substitution of (39) and (41) into (36) yields the following stabilized subsystem:

$$\begin{aligned}\dot{z}_{F_x} &= -c_{F_x} z_{F_x} - (1/(m-3m_a))z_u + z_{p_x} \\ \dot{z}_{F_y} &= -c_{F_y} z_{F_y} - (1/(m-3m_a))z_v + z_{p_y}. \qquad (43)\end{aligned}$$

d) Step 4: For the stabilization of the subsystem (42), we renew the Lyapunov function

$$V_4 = V_3 + (1/2)\big(z_{p_x}^2 + z_{p_y}^2\big). \qquad (44)$$

To have negative terms in $\dot{V}_4$, except for the term $||\boldsymbol{\delta}||^2/4k_1$, the control inputs are selected as

$$\begin{aligned}u_{F_x} &= \tau_1\big(-z_{F_x} + \tau_1^{-1}p_x + \tau_1^{-1}F_x + \dot{p}_{x,\text{des}} - c_{p_x} z_{p_x}\big) \\ u_{F_y} &= \tau_1\big(-z_{F_y} + \tau_1^{-1}p_y + \tau_1^{-1}F_y + \dot{p}_{y,\text{des}} - c_{p_y} z_{p_y}\big). \qquad (45)\end{aligned}$$

Then the derivative of V_4 becomes

$$\dot{V}_4 = \dot{V}_3 - c_{p_x} z_{p_x} - c_{p_y} z_{p_y}. \qquad (46)$$

The substitution of the control inputs u_{F_x} and u_{F_y} into (42) results in the following subsystem:

$$\begin{aligned}\dot{z}_{p_x} &= -c_{p_x} z_{p_x} - z_{F_x} \\ \dot{z}_{p_y} &= -c_{p_y} z_{p_y} - z_{F_y}. \qquad (47)\end{aligned}$$

e) Step 5: To stabilize (20), the desired value for r_e is

$$r_{e,\text{des}} = -c_\psi \psi_e. \qquad (48)$$

The updated Lyapunov function and its derivative are

$$\begin{aligned}V_5 &= V_4 + \frac{1}{2}\psi_e^2 \\ \dot{V}_5 &= \dot{V}_4 - c_\psi \psi_e^2. \qquad (49)\end{aligned}$$

The variable r_e is not a true control so we introduce the following error variable:

$$z_r = r_e - r_{e,\text{des}} \Rightarrow r_e = z_r + r_{e,\text{des}}. \qquad (50)$$

After substitution into (20), we obtain

$$\dot{\psi}_e = -c_\psi \psi_e + z_r. \qquad (51)$$

The time derivative of r_e from (15) is expressed as

$$\dot{r}_e = (1/m_{33})m_z - \dot{r}_R + (1/m_{33})M_z. \qquad (52)$$

Using (48), (50), and (52), the time derivative of z_r is

$$\dot{z}_r = (1/m_{33})m_z - \dot{r}_R + c_\psi r_e + (1/m_{33})M_z. \qquad (53)$$

For the new Lyapunov function, given below, we compute the desired value of M_z

$$\begin{aligned}V_6 &= V_5 + (1/2)z_r^2 \qquad (54) \\ \dot{V}_6 &= \dot{V}_5 + z_r(\psi_e + (1/m_{33})m_z - \dot{r}_R + c_\psi r_e + (1/m_{33})M_z) \qquad (55)\end{aligned}$$

$$M_{z,\text{des}} = m_{33}(-\psi_e - (1/m_{33})m_z + \dot{r}_R - c_\psi r_e - c_r z_r) \qquad (56)$$

where M_z is not a true control, so we introduce the following errors:

$$z_{M_z} = M_z - M_{z,\text{des}} \Rightarrow M_z = z_{M_z} + M_{z,\text{des}}. \qquad (57)$$

The substitution of M_z into (53) yields

$$\dot{z}_r = -c_r z_r - \psi_e + (1/m_{33}) z_{M_z}. \tag{58}$$

The derivative of z_{M_z} is

$$\dot{z}_{M_z} = p_z - \dot{M}_{z,\text{des}}. \tag{59}$$

We again update the Lyapunov function

$$V_7 = V_6 + (1/2) z_{M_z}^2 \tag{60}$$

$$\dot{V}_7 = \dot{V}_6 + z_{M_z}(p_z - \dot{M}_{z,\text{des}} + (1/m_{33}) z_r). \tag{61}$$

The desired value for p_z and the equivalent error variable are

$$p_{z,\text{des}} = \dot{M}_{z,\text{des}} - c_{Mz} z_{Mz} - (1/m_{33}) z_r \tag{62}$$

$$z_{p_z} = p_z - p_{z,\text{des}} \Rightarrow p_z = z_{p_z} + p_{z,\text{des}}. \tag{63}$$

After substitution of (63) into (59), we obtain

$$\dot{z}_{M_z} = -c_{Mz} z_{Mz} - (1/m_{33}) z_r + z_{p_z}. \tag{64}$$

Based on (17), the derivative of z_{p_z} is

$$\dot{z}_{p_z} = -(1/\tau_1) p_z - (1/\tau_1) M_z - \dot{p}_{z,\text{des}} + (1/\tau_1) u_{M_z}. \tag{65}$$

We update the Lyapunov function and we compute u_{Mz}

$$V_8 = V_7 + (1/2) z_{p_z}^2 \tag{66}$$

$$\dot{V}_8 = \dot{V}_7 + z_{p_z}\left(z_{M_z} - \frac{1}{\tau_1} p_z - \frac{1}{\tau_1} M_z - \dot{p}_{z,\text{des}} + \frac{1}{\tau_1} u_{M_z}\right) \tag{67}$$

$$u_{M_z} = \tau_1\left(-z_{M_z} + \tau_1^{-1} p_z + \tau_1^{-1} M_z + \dot{p}_{z,\text{des}} - c_{p_z} z_{p_z}\right). \tag{68}$$

After substitution of u_{Mz} into (65), we have

$$\dot{z}_{M_z} = -c_{pz} z_{pz} - z_{M_z}. \tag{69}$$

The final Lyapunov function is V_8 and will be denoted by V_f. The final stabilized system consists of (28), (37), (43), (47), (51), (64), and (69). The controller is defined by (45) and (68).

f) Step 6: To formally validate the convergence of the controller to a neighborhood of (0, 0), we apply the comparison lemma. We define the errors vector: $\mathbf{w} = [x_e, y_e, \psi_e, z_u, z_v, z_r, z_{Fx}, z_{Fy}, z_{Mz}, z_{px}, z_{py}, z_{pz}]^T$. We consider $g = \min\{k, k_1, c_u, c_v, c_\psi, c_{Fx}, c_{Fy}, c_{Mz}, c_{px}, c_{py}, c_{pz}\}$. Then, the following hold:

$$V_f = \frac{1}{2}\|w\|^2 \tag{70}$$

$$\dot{V}_f \leq -2gV_f + (\|\boldsymbol{\delta}\|^2/4k_1). \tag{71}$$

Following the employment of the comparison lemma, this inequality yields:

$$V_f(t) \leq V_f(0) e^{-2gt} + (\|\boldsymbol{\delta}\|^2/2g4k_1), \quad t \in [0, t_{\text{final}}) \tag{72}$$

$$\|\mathbf{w}(t)\| \leq \|\mathbf{w}(0)\| e^{-gt} + \left(\sqrt{\|\boldsymbol{\delta}\|^2/g4k_1}\right), \quad t \in [0, t_{\text{final}}). \tag{73}$$

The error remains in a neighborhood of zero, which can be reduced as desired by increasing k_1. The three control inputs u_{Fx}, u_{Fy}, and u_{Mz} are directed to the allocation scheme, in the place of F_x, F_y, and M_z, to be resolved into thrust and angle control inputs: $J_{a,\text{des}}$, $J_{b,\text{des}}$, and $J_{c,\text{des}}$ and $\varphi_{a,\text{des}}$, $\varphi_{b,\text{des}}$, and $\varphi_{c,\text{des}}$, respectively. Fig. 7 depicts the control allocation for BS.

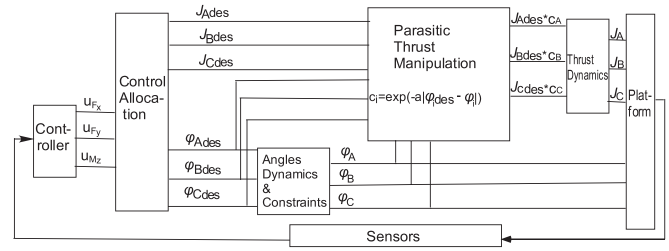

Fig. 7. Allocation diagram BS controller.

In (25), we are using a Lyapunov of the form $V_1 = (1/2)x_e^2 + (1/2)y_e^2$. Even if, we chose a Lyapunov with a nondiagonal positive definite matrix of the form $V = [x_e, y_e]\mathbf{P}[x_e, y_e]^T$, we would change only the speed of convergence of the controller. Since we can achieve any speed of convergence with a diagonal matrix diag(1/2, 1/2) simply by adjusting k (23), the extra entries of P will not provide us more tuning capabilities in order to improve the energy consumption, the accuracy of the dynamic positioning, or the robustness of the platform. This holds for any intermediate Lyapunov function we design in the BS process. Furthermore, using a Lyapunov of the form $V = [x_e, y_e]\mathbf{P}[x_e, y_e]^T$ does not allow us to use the comparison lemma (70)–(73) in order to formally validate the convergence of the controller.

3) Tuning Procedure: The final stabilized system can be written in matrix form

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{D} \tag{74}$$

where $\mathbf{x} = [x_e, y_e, \psi_e, z_u, z_v, z_r, z_{F_x}, z_{F_y}, z_{M_z}, z_{p_x}, z_{p_y}, z_{p_z}]^T$ and $\mathbf{A}$ is a square matrix 12×12. Diagonal entries can be adjusted during the tuning process: $a_{1,1} = -(k + k_1)$, $a_{2,2} = -(k + k_1)$, $a_{3,3} = -c_\psi$, $a_{4,4} = -c_u$, $a_{5,5} = -c_v$, $a_{6,6} = -c_{\text{r}}$, $a_{7,7} = -c_{\text{Fx}}$, $a_{8,8} = -c_{\text{Fy}}$, $a_{9,9} = -c_{\text{Mz}}$, $a_{10,10} = -c_{\text{px}}$, $a_{11,11} = -c_{\text{py}}$, and $a_{12,12} = -c_{\text{pz}}$. Diagonal terms must be chosen so that the eigenvalues of $\mathbf{A}$ have negative real parts. $\mathbf{D} = [\boldsymbol{\delta}^T|\mathbf{0}]^T$. The real part of the eigenvalues determines the rate of convergence of the errors to zero. The mass of the platform, hardware limitations, and settling delays limit the convergence rate and the gains cannot increase arbitrarily.

The platform carries heavy loads so the mass is not exactly known. The velocities and the forces are not measured. So, this multistep BS provides robustness to our controller and more tolerance to the error propagation.

B. Model Predictive Controller

In a preliminary study of an MPC [21], we found that when a large prediction horizon was applied, it was observed that a slight change in the state values resulted in considerable changes to the computed control input, i.e., an undesired property for systems with limitations. Here, we present an MPC, in which the large prediction horizon issue is handled effectively. Therefore, here large prediction horizons can be used in the tuning process, providing extended capabilities to the MPC.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

In the design of the MB-PID [19] and of the MPC [21], initially the model-based technique is employed. Forces and torque are chosen such that nonlinearities are canceled. This yields a linear system with bounded disturbances, which can be regulated by a PID controller or by a linear MPC. We compute $\dot{x}(t)$ by subtracting two consecutive values of x and then divide them by the time step of the GPS. We compute $\ddot{x}$ by subtracting two consecutive values of $\dot{x}$ and then divide by the time step of the GPS. Position is provided through GPS and orientation through a compass system.

1) Model-Based Technique: After derivation of the kinematic equation (4), we obtain

$$\ddot{\mathbf{x}} = \mathbf{R}\dot{\mathbf{v}} + \dot{\mathbf{R}}\mathbf{v} \Leftrightarrow \dot{\mathbf{v}} = \mathbf{R}^{-1}(\ddot{\mathbf{x}} - \dot{\mathbf{R}}\mathbf{v}). \tag{75}$$

Substitution of (75) into the equation of motion (5) yields

$$\boldsymbol{\tau}_c = \mathbf{M}\mathbf{R}^{-1}\ddot{\mathbf{x}} - \mathbf{M}\mathbf{R}^{-1}\dot{\mathbf{R}}\mathbf{R}^{-1}\dot{\mathbf{x}} - \mathbf{q} - \mathbf{q}_{\text{dist}}. \tag{76}$$

To reduce our system under control to a linear system with bounded disturbances, we select the following $\boldsymbol{\tau}_{c,\text{des}}$:

$$\boldsymbol{\tau}_{c,\text{des}} = \mathbf{M}\mathbf{R}^{-1}\mathbf{f}_{\text{fb}} - \mathbf{M}\mathbf{R}^{-1}\dot{\mathbf{R}}R^{-1}\dot{\mathbf{x}} - \mathbf{q}. \tag{77}$$

After substitution of $\boldsymbol{\tau}_{c,\text{des}}$ (77) into $\boldsymbol{\tau}_c$ (76), the following linear system with bounded disturbances results:

$$\ddot{\mathbf{x}} = \mathbf{f}_{\text{fb}} - \mathbf{R}\mathbf{M}^{-1}\mathbf{q}_{\text{dist}} \tag{78}$$

where vector $\mathbf{f}_{\text{fb}(3\times1)} = [u_{Fx}, u_{Fy}, u_{Mz}]^T$ is the control input.

The double integrator (78) can be rewritten as

$$\begin{aligned}\dot{\mathbf{x}}_l &= \begin{bmatrix} \mathbf{0}_{3\times3} & \mathbf{I}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \end{bmatrix}\mathbf{x}_l + \begin{bmatrix} \mathbf{0}_{3\times3} \\ \mathbf{I}_{3\times3} \end{bmatrix}\mathbf{f}_{\text{fb}} + \begin{bmatrix} \mathbf{0}_{3\times1} \\ -\mathbf{R}\mathbf{M}^{-1}\mathbf{q}_{\text{dist}} \end{bmatrix} \\ \mathbf{y} &= \begin{bmatrix} \mathbf{I}_{3\times3} & \mathbf{0}_{3\times3} \end{bmatrix}\mathbf{x}_l \end{aligned} \tag{79}$$

where $\mathbf{x}_l = [x, y, \psi, \dot{x}, \dot{y}, \dot{\psi}]^T$.

The vector $\mathbf{R}\mathbf{M}^{-1}\mathbf{q}_{\text{dist}}$ corresponds to the external bounded disturbance. The eigenvalues of (79) are all equal to zero, so our reduced system is not asymptotically stable. This is the reason why, a large prediction horizon leads to an ill-conditioned optimization problem.

2) Augmented Model: We introduce an augmented description of the linear system (79). The augmented state vector consists of the time derivative of the state vector in (79) augmented by the three observed outputs x, y, and ψ

$$\begin{aligned}\dot{\mathbf{x}}_{\text{aug}} &= \begin{bmatrix} \mathbf{0}_{3\times3} & \mathbf{I}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \\ \mathbf{I}_{3\times3} & \mathbf{0}_{3\times3} & \mathbf{0}_{3\times3} \end{bmatrix}\mathbf{x}_{\text{aug}} + \begin{bmatrix} \mathbf{0}_{3\times3} \\ \mathbf{I}_{3\times3} \\ \mathbf{0}_{3\times3} \end{bmatrix}\dot{\mathbf{f}}_{\text{fb}} \\ &= \mathbf{A}\mathbf{x}_{\text{aug}} + \mathbf{B}\dot{\mathbf{f}}_{\text{fb}} \\ \mathbf{y}_{\text{aug}} &= \begin{bmatrix} \mathbf{0}_{3\times3} & \mathbf{I}_{3\times3} \end{bmatrix}\mathbf{x}_{\text{aug}} \end{aligned} \tag{80}$$

where $\mathbf{x}_{\text{aug}} = [\dot{x}, \dot{y}, \dot{\psi}, \ddot{x}, \ddot{y}, \ddot{\psi}, x, y, \psi]^T$. Vector $\mathbf{R}\mathbf{M}^{-1}\mathbf{q}_{\text{dist}}$ is treated as an external bounded disturbance, the control has no information about $\mathbf{q}_{\text{dist}}$; that is why its derivative does not appear in (80). The appearance of $\dot{\mathbf{f}}_{\text{fb}}$ as the control input is essential, since the quadratic cost function of the MPC contains $\dot{\mathbf{f}}_{\text{fb}}$.

The augmented state vector contains the state vector of the linear system (79), as well as the second derivative of the position and orientation variables. By regulating the augmented model, we also regulate the linear model (79). So, the augmented model (80) will be used as system model instead of (79). Note that the eigenvalues of (80) are all equal to zero, so it is not asymptotically stable.

3) Receding Horizon Principle: In the developed MPC, we compute the prediction of the state vector and the optimal control input trajectory within a prediction horizon. After the computation of the optimal trajectory of the derivative of $\mathbf{f}_{\text{fb}}$, we integrate it, we feed the plant with the initial value of the control input only, and the plant performs a small motion. Afterward, we compute again the prediction of the state and the optimal control trajectory, having as initial state the new state of the plant. Our new prediction horizon is the previous one, receded by one time instant. This procedure is repeated until the plant reaches the desired position and orientation. The use of the initial value of $\dot{\mathbf{f}}_{\text{fb}}$ led to the idea that the derivative of the control can be a state feedback

$$\dot{\mathbf{f}}_{\text{fb}}(t) = -\mathbf{K}_{\text{mpc}}\mathbf{x}_{\text{aug}}(t). \tag{81}$$

We focus on the computation of matrix $\mathbf{K}_{\text{mpc}}$, which will come of the optimization phase. In order to simplify our computations and save memory space in our computing systems, we approximate the time derivative of $\mathbf{f}_{\text{fb}}$ with a linear combination of Laguerre functions

$$\dot{\mathbf{f}}_{\text{fb}}(t) \approx \begin{bmatrix} \mathbf{L}_1(t)^T\boldsymbol{\eta}_1 \\ \mathbf{L}_2(t)^T\boldsymbol{\eta}_2 \\ \mathbf{L}_3(t)^T\boldsymbol{\eta}_3 \end{bmatrix} \tag{82}$$

where $\mathbf{L}_j = [l_1, \ldots, l_{N_j}]^T$ and $\boldsymbol{\eta}_j = [c_1, \ldots, c_{N_j}]^T$, $j = 1, 2, 3$.

The upper bound for index j is equal to the size of control vector $\mathbf{f}_{\text{fb}}$. The number of Laguerre functions used in the approximation is given by N_j. Vector $\eta_{\mathbf{j}}$ contains the coefficients of the linear combination. Laguerre functions are defined by

$$l_i(t) = \sqrt{2p_j}\frac{e^{p_j t}}{(i-1)!}\frac{d^{i-1}}{dt}[t^{i-1}e^{-2p_j t}]. \tag{83}$$

The coefficient p_j is the time scaling factor for the Laguerre functions participating in L_j, which determines the rate of their exponential decay. N_j and p are design parameters determined during the tuning procedure.

4) Cost Function: To shift the eigenvalues of (80) to the left-half plane, we use the technique of exponential data weighting. To this end, we introduce the following cost function:

$$\begin{aligned}\mathbf{J}_1 = \int_0^{T_p} \Big(&e^{-2a\tau}\mathbf{x}_{\text{aug}}(t_i+\tau|t_i)^T\mathbf{Q}\mathbf{x}_{\text{aug}}(t_i+\tau|t_i) \\ &+ e^{-2a\tau}\dot{\mathbf{f}}_{\text{fb}}^T\mathbf{W}\dot{\mathbf{f}}_{\text{fb}}\Big)d\tau \end{aligned} \tag{84}$$

subject to (80)

$$\dot{\mathbf{x}}_{\text{aug}}(t_i+\tau|t_i) = \mathbf{A}\mathbf{x}_{\text{aug}}(t_i+\tau|t_i) + \mathbf{B}\dot{\mathbf{f}}_{\text{fb}}(\tau), \quad 0 \le \tau \le \mathrm{T}_p. \tag{85}$$

The cost function in (84) includes the derivative of the control input $\mathbf{f}_{\text{fb}}$ and $\mathbf{x}_{\text{aug}}(t_i+\tau|t_i)$ is the prediction of the state vector of the augmented model given by (85), for the time interval $[t_i, t_i + T_p]$ [see (91)]. $\mathbf{Q}$ and $\mathbf{W}$ are diagonal positive definite matrices, T_p is the prediction horizon, and $a > 0$ is the exponential decay factor. These four design parameters are determined during the tuning procedure.

If we make the following change of variables:

$$\mathbf{x}_{\mathrm{aug},a}(t_i+\tau|t_i) = e^{-a\tau}\mathbf{x}_{\mathrm{aug}}(t_i+\tau|t_i), \quad \dot{\mathbf{f}}_{\mathrm{fb},a}(\tau) = e^{-a\tau}\dot{\mathbf{f}}_{\mathrm{fb}}(\tau) \tag{86}$$

$$\mathbf{x}_{\mathrm{aug},a}(t_i|t_i) = \mathbf{x}_{\mathrm{aug}}(t_i|t_i), \quad \dot{\mathbf{f}}_{\mathrm{fb},a}(0) = \dot{\mathbf{f}}_{\mathrm{fb}}(0), \quad \text{for } \tau = 0 \tag{87}$$

minimization of (84) subject to (85) is equivalent to

$$\mathbf{J} = \int_0^{T_p} \big(\mathbf{x}_{\mathrm{aug},a}(t_i+\tau|t_i)^T \mathbf{Q}\mathbf{x}_{\mathrm{aug},a}(t_i+\tau|t_i) + \dot{\mathbf{f}}_{\mathrm{fb},a}^T \mathbf{W}\dot{\mathbf{f}}_{\mathrm{fb},a}\big) d\tau \tag{88}$$

subject to

$$\dot{\mathbf{x}}_{\mathrm{aug},a}(t_i+\tau|t_i) = \mathbf{A}_a\mathbf{x}_{\mathrm{aug},a}(t_i+\tau|t_i) + \mathbf{B}\dot{\mathbf{f}}_{\mathrm{fb},a}(\tau), \quad 0 \le \tau \le \mathrm{T}_p \tag{89}$$

where $\mathbf{A}_a = \mathbf{A} - a\mathbf{I}$.

The eigenvalues of $\mathbf{A}_a$ (89) are the eigenvalues of $\mathbf{A}$ (85) shifted by $-a$, where $a > 0$ is the exponential decay factor. Since these two minimization problems are equivalent, instead of applying the MPC technique to (84) and (85), we can apply it to (88) and (89). Note that, although the state vector $\mathbf{x}_{\mathrm{aug},a}(\cdot)$ and the derivative of the control input $\dot{\mathbf{f}}_{\mathrm{fb},a}(\cdot)$ in (89) are different from the state vector $\mathbf{x}_{\mathrm{aug}}(\cdot)$ and the control input $\dot{\mathbf{f}}_{\mathrm{fb}}(\cdot)$ in (85), their initial values are equal [see (87)]. Therefore, the initial control values we provide to the system (80) do not change, while the ill-conditioning issue is resolved. From (81) and (86), it also holds

$$\dot{\mathbf{f}}_{\mathrm{fb},a}(t) = -\mathbf{K}_{\mathrm{mpc}}\mathbf{x}_{\mathrm{aug},a}(t). \tag{90}$$

5) Prediction Phase: The cost function (88) includes the prediction of the state vector over the prediction horizon. This can be achieved by solving the linear system (89), for $0 \le \tau \le T_p$. Therefore, we substitute (82) into (86) and then into (89) to solve and obtain

$$\mathbf{x}_{\mathrm{aug},a}(t_i+\tau|t_i) = e^{\mathbf{A}_a\tau}\mathbf{x}_{\mathrm{aug},a}(t_i) + \boldsymbol{\varphi}(\tau)^T\boldsymbol{\eta}, \quad 0 \le \tau \le T_p \tag{91}$$

where

$$\begin{aligned}\boldsymbol{\varphi}(\tau)^{\mathrm{T}} &= [\boldsymbol{\varphi}_1(\tau)^T, \boldsymbol{\varphi}_2(\tau)^T, \boldsymbol{\varphi}_3(\tau)^T]\\ \boldsymbol{\varphi}_j(\tau)^T &= \int_0^{\tau} e^{\mathbf{A}_a(\tau-\gamma)}\mathbf{B}_j\mathbf{L}_j(\gamma)^T d\gamma \quad j = 1,2,3.\end{aligned} \tag{92}$$

The dimension of $\boldsymbol{\varphi}_j(\tau)$ is equal to $n \times N_j$, $i = 1, 2, 3$, where n is the dimension of $\mathbf{x}_{\mathrm{aug},a}(\cdot)$. The matrix $\boldsymbol{\varphi}_j(\tau)^T$ can be computed as the solution of the following equation:

$$\mathbf{A}_a\boldsymbol{\varphi}_j(\tau)^T - \boldsymbol{\varphi}_j(\tau)^T\mathbf{A}_{p_j}^T = -\mathbf{B}_j\mathbf{L}_j(\tau)^T + e^{\mathbf{A}_a\tau}\mathbf{B}_j\mathbf{L}_j(0)^T \tag{93}$$

where

$$\mathbf{A}_{p_j} = \begin{bmatrix} -p_j & 0 & \cdots & 0 \\ -2p_j & -p_j & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ -2p_j & \cdots & -2p_j & -p_j \end{bmatrix}. \tag{94}$$

The dimension of $\mathbf{A}_{p_j}$ is $N_j \times N_j$. The coefficients $\boldsymbol{\eta} = [\boldsymbol{\eta}_1^T, \boldsymbol{\eta}_2^T, \boldsymbol{\eta}_3^T]^T$ will be computed in the optimization phase.

6) Optimization Phase: Substituting (90) and (91) into (88), we compute the optimal coefficients $\boldsymbol{\eta}$ that minimize the cost function (88) for $0 \le \tau \le T_p$

$$\boldsymbol{\eta} = -\boldsymbol{\Omega}^{-1}\boldsymbol{\Psi}\mathbf{x}_{\mathrm{aug},a}(t_i|t_i) \tag{95}$$

where

$$\boldsymbol{\Omega} = \int_0^{T_p} \boldsymbol{\varphi}(\tau)\mathbf{Q}\boldsymbol{\varphi}(\tau)^T d\tau + \mathbf{W}_L, \boldsymbol{\Psi} = \int_0^{T_p} \boldsymbol{\varphi}(\tau)\mathbf{Q}e^{\mathbf{A}_a\tau} d\tau \tag{96}$$

where $\mathbf{W} = \mathrm{diag}\{w_1, w_2, w_3\}$ and $\mathbf{W}_L$ is a block diagonal matrix with the jth block being $\mathbf{W}_j = w_j\mathbf{I}_{Nj\times Nj}$, $j = 1, 2, 3$. From (87) and (95), it holds

$$\boldsymbol{\eta} = -\boldsymbol{\Omega}^{-1}\boldsymbol{\Psi}\mathbf{x}_{\mathrm{aug}}(t_i|t_i). \tag{97}$$

Using (82) and (86), the trajectory of the derivative of the control input is

$$\dot{\mathbf{f}}_{\mathrm{fb},a}(\tau) = e^{-a\tau}\begin{bmatrix} L_1(\tau)^T & 0 & 0 \\ 0 & L_2(\tau)^T & 0 \\ 0 & 0 & L_3(\tau)^T \end{bmatrix}\begin{bmatrix}\boldsymbol{\eta}_1\\ \boldsymbol{\eta}_2\\ \boldsymbol{\eta}_3\end{bmatrix}. \tag{98}$$

In each cycle of computation, we keep only the initial value of the whole trajectory. After substitution of (97) into (98) and according to (87), we obtain

$$\dot{\mathbf{f}}_{\mathrm{fb}}(0)_{t=t_i} = \begin{bmatrix} \mathbf{L}_1(0)^T & 0 & 0 \\ 0 & \mathbf{L}_2(0)^T & 0 \\ 0 & 0 & \mathbf{L}_3(0)^T \end{bmatrix}\boldsymbol{\Omega}^{-1}\boldsymbol{\Psi}\mathbf{x}_{\mathrm{aug}}(t_i|t_i) \tag{99}$$

$$\mathbf{K}_{\mathrm{mpc}} = \begin{bmatrix} \mathbf{L}_1(0)^T & 0 & 0 \\ 0 & \mathbf{L}_2(0)^T & 0 \\ 0 & 0 & \mathbf{L}_3(0)^T \end{bmatrix}\boldsymbol{\Omega}^{-1}\boldsymbol{\Psi}. \tag{100}$$

Methodologies for embedding the constraints in the algorithm of the MPC exist. However, the constrained quadratic programming optimization problem, which produces the solution, is computationally intensive. In the proposed MPC, $\mathbf{K}_{\mathrm{mpc}}$ is computed once. In the methodologies where constraints are embedded, $\mathbf{K}_{\mathrm{mpc}}$ must be computed in every control loop, adding significant time delays, which may lead to loss of control. Instead of embedding the constraints, we penalize $\dot{\mathbf{f}}_{\mathrm{fb}}$ with appropriate tuning of $\mathbf{W}$. Now, we have $\mathbf{f}_{\mathrm{fb}}$. Using (77), we can compute $\boldsymbol{\tau}_{c,\mathrm{des}} = [F_{x,\mathrm{des}}, F_{y,\mathrm{des}}, M_{z,\mathrm{des}}]^T$. Using the allocation scheme, we resolve the desired forces/torque to the signals of desired thrusts/angles which will be driven to the actuators (Fig. 8).

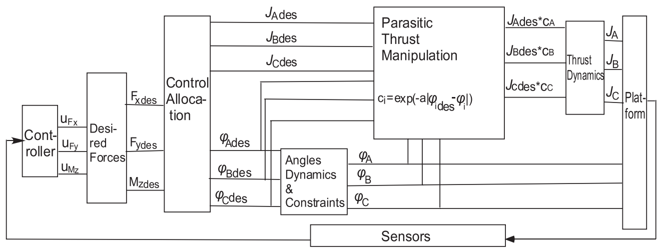

Fig. 8. Allocation diagram MPC.

7) Tuning Procedure: The exponential decay factor $a > 0$ must be positive enough so that all eigenvalues have negative real parts. The prediction horizon T_p is selected sufficiently large to capture the evolution of the state vector. The weight matrices $\mathbf{Q}$ and $\mathbf{W}$ must be tuned so that the platform reaches the target, overcoming the environmental disturbances, without exceeding thruster limits. Two or three Laguerre functions are adequate for the approximation of the $\dot{\mathbf{f}}_{\mathrm{fb}}$ around a neighborhood of the initial value. Laguerre decay coefficients p_j are not given high values to avoid zero input at the very beginning of the prediction horizon. Fig. 8 depicts the control allocation of the platform in case of the MPC.

V. Evaluation of the Controllers

The goal is the platform to enter a *target circle* centered at (0, 0) with radius $r = 5$ m and to be directed to a predefined direction of 0°. We evaluate the performance of the controller while the platform performs Task A and Task B.

In Task A, the platform starts from an initial position $(-20, -20)$ with an initial orientation of $-20°$. It has to enter and remain inside the *target circle* at 0° with a tolerance of $\pm 10°$ in the presence of environmental disturbances. In **Task B**, the platform is placed at (0, 0) and at an orientation of 5°; it must remain inside the *target circle* at an orientation of 0° with a tolerance $\pm 10°$. In Task B, the controllers were tuned so that they exhibit the same maximum displacement from (0, 0) under the same disturbances.

The evaluation is based on three criteria: 1) energy consumption; 2) accuracy of the dynamic positioning; and 3) robustness in mass changes.

1) The energy consumed (E) by the platform is computed using a curve, which relates the thrust force to the input power needed to produce this thrust. The curve is provided by the manufacturer of the platform jets and is introduced as a lookup table in the MATLAB/Simulink simulation.
2) The accuracy of the dynamic positioning is measured by

$$G = \int_0^T e(t)dt \tag{101}$$

where

$$e = \sqrt{(x - x_{\mathrm{des}})^2 + (y - y_{\mathrm{des}})^2 + (\psi - \psi_{\mathrm{des}})^2}. \tag{102}$$

The smaller the time integral of this error, the better the accuracy of the dynamic positioning.

TABLE I
MB-PID Tuning, Task A

k_d=0.12	k_p=0.0048	k_i=0.000064

TABLE II
BS Tuning, Task A

k=0.07	k_1=0.07	c_u=0.07	c_v=0.07
c_{Fx}=0.07	c_{Fy}=0.07	c_ψ=1.2	c_r=0.7
c_{Mz}=0.7	c_{px}=1.5	c_{py}=1.5	c_{pz}=1.5

Especially, for the BS controller for Task A, the reference position and orientation x_r, y_r, and ψ_r are given by a fifth-order time polynomial with an initial value of -20 and a final value of 0. Since $x_r(0) = -20$, $x_r(T_{\text{final}}) = 0$ and $y_r(0) = -20$, $y_r(T_{\text{final}}) = 0$, the fifth-order time polynomial of x_r is the same as that of y_r, $x_r = y_r$. Therefore, (x_r, y_r) is a straight line connecting $(-20, -20)$ with $(0, 0)$. For Task B, x_r and y_r are set to zero and $\psi_r = 5°$. The u_r, v_r, and r_r are computed after time differentiation of the polynomials of x_r, y_r, and ψ_r and then use of (4). The other two controllers do not follow a trajectory. Our goal is to design controllers, which provide the best possible accuracy of dynamic positioning, while they consume the least possible energy in given time T. To correlate the accuracy of dynamic positioning with the energy consumed, we introduce the PI characterizing controller performance

$$\mathrm{PI} = (E/T) \cdot (G/T) \tag{103}$$

where E is the energy consumed and T is the total time of the task.

3) Platforms like Vereniki are used to carry loads to be submerged, which change considerably the total mass of the platform. It is important for the controllers to endure large errors in the change of mass.

A. Comparative Study—Task A

1) Presentation of the Results: In this section, we are presenting the results employing the controllers in Task A. The simulation lasts for $T = 600$ s. The gains of the MB-PID were chosen after trial and error in a way that the consumed energy is kept low and the platform remains inside the circle despite the environmental disturbances (see [19], [22]). The results of the tuning process of the MB-PID controller are given in Table I.

For the BS controller, the time constants are $\tau_1 = \tau_2 = \tau_3 = 50$ s. Considerably, smaller time constants result in inadequate handling of the delays, while a higher time constant makes the response very slow. The gains of the BS controller are given in Table II. Much higher gains result in high convergence rate above the capabilities of the platform.

Table III presents the $\mathbf{Q}$ and $\mathbf{W}$ matrices for the MPC. Very high values in the diagonal elements of $\mathbf{Q}$ result in thrust

TABLE III
MPC **Q**-**W** TUNING, TASK A

$\mathbf{Q}=\mathbf{I}_{9\times 9}$	$\mathbf{W}=10^{8}\mathbf{I}_{3\times 3}$

TABLE IV
MPC LAGUERRE PARAMETERS TUNING, TASK A

	p	N
i=1	2	2
i=2	2	2
i=3	2	2

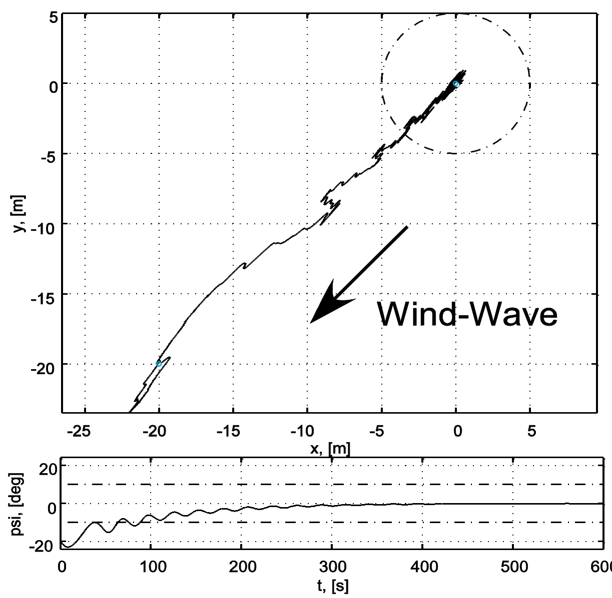

Fig. 9. Model predictive controller dynamic positioning—Task A.

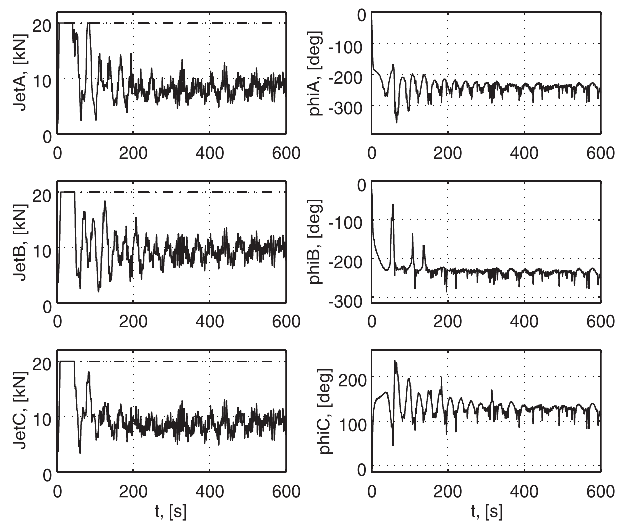

Fig. 10. Model predictive controller jet thrusts/angles—Task A.

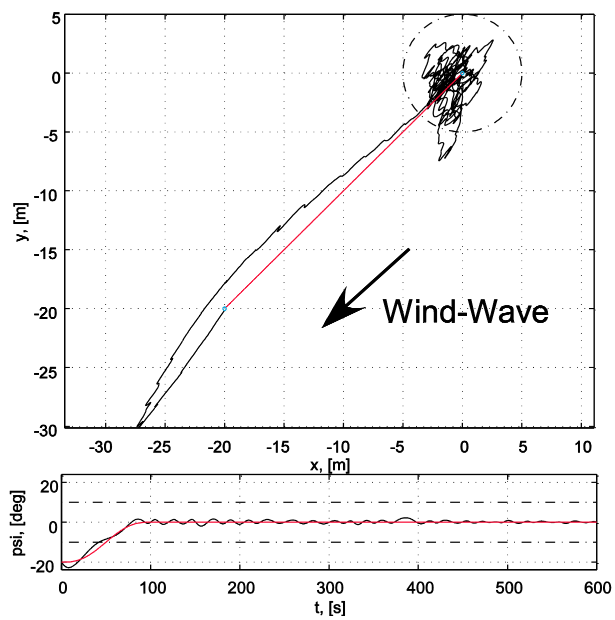

Fig. 11. BS controller dynamic positioning—Task A.

requirements above their limits. Much smaller values in the diagonal elements of **W** result in control inputs that exceed the limits. Very high values in the diagonal elements of **W** keep the control inputs low; however, in such a case, the platform cannot overcome the environmental disturbances.

Table IV includes the Laguerre decay coefficients and the number of Laguerre functions for each control input. Much higher values in p_i result in immediate convergence of the control input to zero. A large number of Laguerre functions would increase drastically the computation time. The coefficient of the exponential data weighting is $a = -0.00001$. A higher a coefficient would increase the convergence rate of the position and orientation to levels not supported by the platform.

In Fig. 9, we observe that the MPC drives the platform to its target and keeps it there with remarkable precision. The platform is also directed to a desired orientation of 0°. Inside the target circle, jet average thrust is around 10 kN, while they are directed at a constant angle each (see Fig. 10).

In Fig. 11, we observe that BS controller also drives the platform to its target, keeps it there, and directs it to a predefined orientation of 0°. The red line on the 2-D figure and the red curve on the ψ figure depict the desired trajectory and the desired orientation, respectively. As shown in Fig. 12, thrusts take high values, while jets rotate around continuously in order to reduce the positioning error.

In Fig. 13, we observe that the MB-PID controller exhibits a high overshoot before finally reaching the target and remaining permanently inside. The overshoot in the MB-PID occurs because to keep the platform inside the circle despite the environmental disturbances, relatively high gains were used. These gains together with the controller integrator and high speeds result in position overshoots. To reduce the overshoot, the gains must be reduced, also reducing the positioning accuracy G, which is undesirable. After the platform reaches the target permanently, jet average thrust remains below 10 kN and is directed at some constant angle each (see Fig. 14).

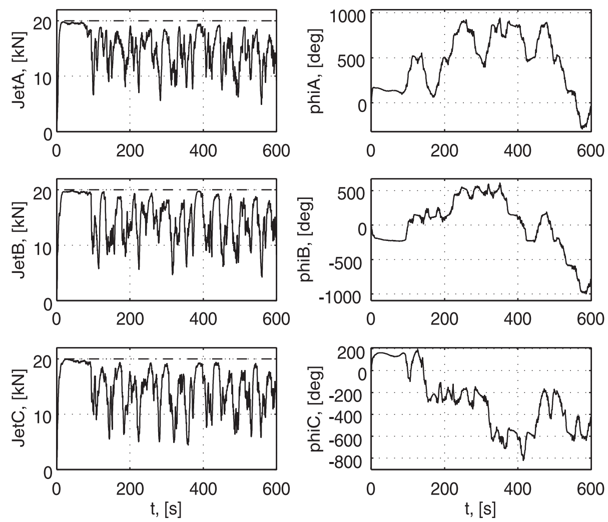

Fig. 12. BS controller jet thrusts/angles—Task A.

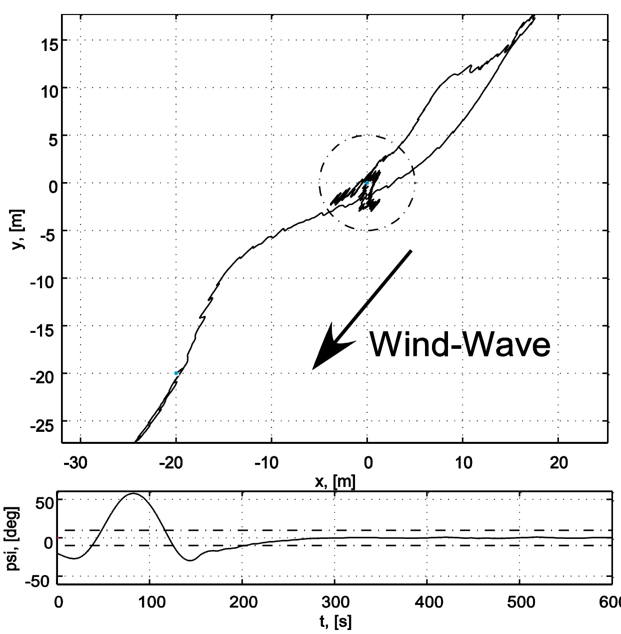

Fig. 13. MB-PID controller dynamic positioning—Task A.

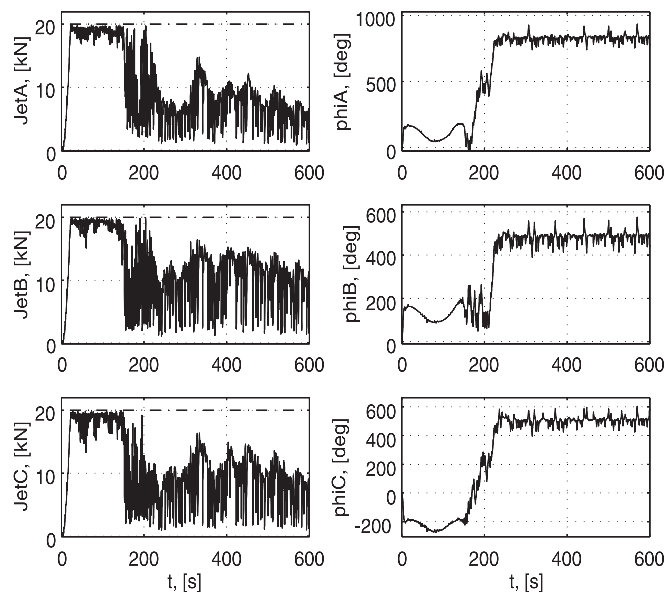

Fig. 14. MB-PID controller jet thrusts/angles—Task A.

TABLE V
PERFORMANCE FOR TASK A

Controller	***G*** (m×s)	***E*** (J)	***PI*** (J×m/s)
BS	2086	3.021e+05	1.750e+03
MPC	1472	1.628e+05	0.665e+03
MB-PID	3483	1.929e+05	1.866e+03

TABLE VI
ENERGY CONSUMPTION, TASK A

Controller	***E*** (J) (parasitic thrust reduction)	***E*** (J) (no parasitic thrust reduction)	Difference(%)
BS	3.021e+05	4.605e+05	-34%
MPC	1.628e+05	1.662e+05	-2%
MB-PID	1.929e+05	2.437e+05	-21%

Table V summarizes the performance of the controllers. The PI of the MPC is considerably lower compared with the other two. The BS controller follows with better dynamic positioning compared with the MB-PID, but with the highest energy consumption.

To examine controller robustness in mass errors, the platform mass is increased by 10%, 30%, 50%, 70%, and 90%, and the changes in energy consumption (E) and dynamic positioning accuracy (G) are noted. We consider the controller that does not exhibit considerable changes in E and G quantities while coping with large errors as robust.

Table VI shows the change in energy consumption when the technique for parasitic thrust reduction is used (see Figs. 4 and 5). Controllers consume less energy when the technique is employed. It affects especially the BS because jet angles do not converge to constant values and there is always a considerable difference between current and desired angle.

In Table VII, the robustness results of the MPC are presented. We observe that even with a 90% increase in mass, the energy consumption of the MPC remains at low levels and it is lower compared with the energy consumption of the BS with 0% increase and of the MB-PID, with a 10% increase. Furthermore, we observe that despite a 90% increase in mass, the G quantity of the MPC is lower compared with the G quantities of BS and MB-PID, even with a 0% increase in mass. The MPC exhibits the lowest PI, during the whole examination process. This remarkable performance characterizes MPC as the most robust controller compared with the other two.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

TABLE VII
ROBUSTNESS RESULTS—MPC—TASK A

Increase in mass (%)	0%	10%	30%	50%	70%	90%
***E* (J×(e+05))**	1.628	1.679	1.715	1.842	1.992	2.155
***G* (m×s)**	1472	1463	1447	1460	1519	1600
***PI* (m× J/s)**	665.66	682.3	689.33	747.02	840.5	957.77

TABLE VIII
ROBUSTNESS RESULTS—BS—TASK A

Increase in mass (%)	0%	10%	30%	50%	70%	90%
***E* (J×(e+05))**	3.021	3.024	3.324	3.423	3.573	3.759
***G* (m×s)**	2086	2039	2248	2349	2445	2614
***PI* (m× J/s)**	1750.3	1712.7	2075.6	2233.5	2426.6	2729.4

TABLE IX
ROBUSTNESS RESULTS—MB-PID—TASK A

Increase in mass (%)	0%	10%	30%	50%	70%	90%
***E* (J×(e+05))**	1.929	2.410	2.380	2.393	2.471	2.487
***G* (m×s)**	3483	3315	3250	3384	3551	3723
***PI* (m× J/s)**	1866.3	2219.1	2148.6	2249.4	2437.3	2571.9

Table VIII presents the robustness results of the BS controller. The energy consumption considerably increases. On the other hand, the G of BS with 90% increase is lower to that of MB-PID with 0% increase.

Table IX summarizes the robustness results of the MB-PID controller. The quantity G is very high due to the overshoot (see Fig. 13), which cannot be eliminated whatever the tuning is. Note that the energy consumption, with a 90% increase in mass, is lower to that of the BS with 0% increase.

In some cases, we observe that an increase in the mass may result in a decrease in G. Increase in the mass results in a smaller displacement of the platform from the initial position and therefore in a smaller G. This cannot be interpreted as improvement in the accuracy of the dynamic positioning. The energy is increasing as the mass increases because thrusters have to drive a higher load.

2) Conclusion—Task A: For Task A, we would definitely propose MPC because it has considerably smaller PI compared with the other two (see Table V). Table X summarizes the results for Task A. As a second choice, we select BS because it has the second lowest PI. The MB-PID has the second lowest energy consumption.

TABLE X
PERFORMANCE FOR TASK A

Controller	***G***	***E***	Robustness
BS	0	-	0
MPC	+	+	+
MB-PID	-	0	0

+ : "very good", 0 : "good", - : " not so good"

TABLE XI
MB-PID TUNING, TASK B

k_d=0.093	k_p=0.0028	k_i=0.000029

TABLE XII
BS TUNING, TASK B

k=0.07	k_1=0.07	c_u=0.07	c_v=0.07
c_{Fx}=0.07	c_{Fy}=0.07	c_ψ=1.2	c_r=0.7
c_{Mz}=0.7	c_{px}=1.5	c_{py}=1.5	c_{pz}=1.5

TABLE XIII
MPC **Q**-**W** TUNING, TASK B

$\mathbf{Q}=0.7\mathbf{I}_{9\times9}$	$\mathbf{W} = diag(3.4, 3.4, 0.8)*10^8$

TABLE XIV
MPC LAGUERRE PARAMETERS TUNING, TASK B

	p	N
i=1	2	2
i=2	2	2
i=3	1	2

B. Comparative Study—Task B

1) Presentation of the Results: For Task B, the simulation for the task runs for $T = 600$ s. The controllers were tuned so that they have the same maximum displacement from the center $d_{\max} = 13$ m. The results of the tuning process of the MB-PID controller [19], [22] are summarized in Table XI.

The gains of the BS controller are summarized in Table XII. The time constants are $\tau_1 = \tau_2 = \tau_3 = 50$ s.

The results of the tuning process of the MPC are provided in Tables XIII and XIV. In the simulation, the value of a, we use is $a = -0.00001$.

As can be seen in Fig. 15, although the disturbances displace the platform, the MPC drives platform inside the target and redirects it to 0°. Inside the target circle, the average thrust is below 10 kN, with jets directed at a constant angle each (see Fig. 16).

In Fig. 17, we observe that BS achieves better dynamic positioning compared with the model-based one. However, the BS response has high energy requirements due to high thrust (see Fig. 18).

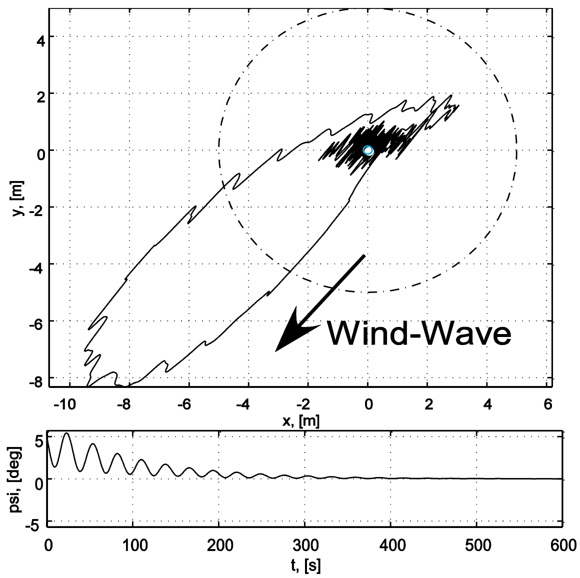

Fig. 15. Model predictive controller dynamic positioning—Task B.

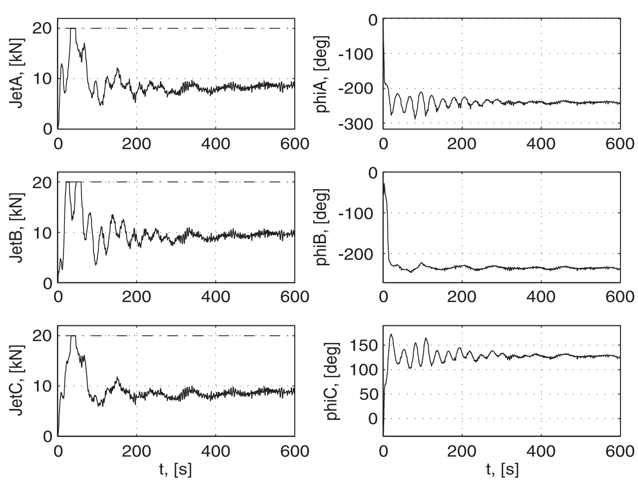

Fig. 16. Model predictive controller jet thrusts/angles—Task B.

TABLE XV
PERFORMANCE FOR TASK B

Controller	***G*** (m×s)	***E*** (J)	***PI*** (J×m/s)
BS	1682	2.949e+05	1.377e+03
MPC	1128	1.498e+05	0.469e+03
MB-PID	1839	1.355e+05	0.691e+03

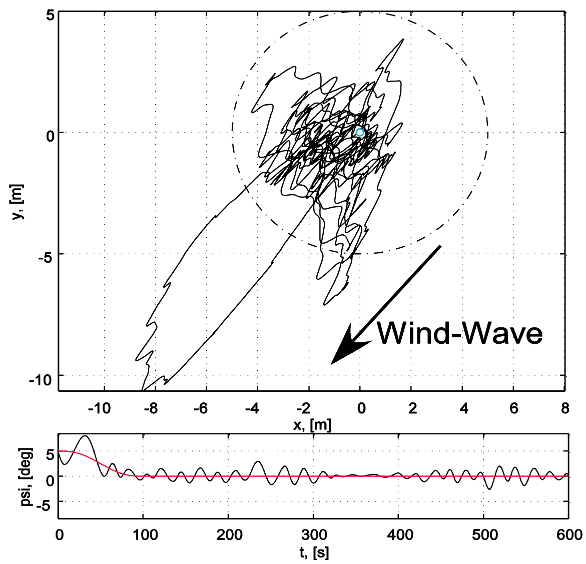

Fig. 17. BS controller dynamic positioning—Task B.

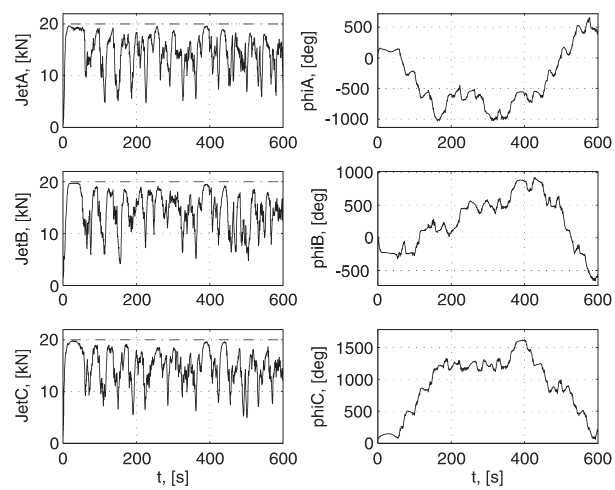

Fig. 18. BS controller jet thrusts/angles—Task B.

TABLE XVI
ENERGY CONSUMPTION, TASK B

Controller	***E*** (J) (parasitic thrust reduction)	***E*** (J) (no parasitic thrust reduction)	Difference(%)
BS	2.949e+05	4.605e+05	-35%
MPC	1.498e+05	1.499e+05	-0.07%
MB-PID	1.355e+05	1.461e+05	-7.3%

In Fig. 19, note that the MB-PID redirects the platform inside the target circle, and to the desired orientation, despite the environmental disturbances. The dynamic positioning is considerably worse compared with MPC; a fact illustrated. On the other hand, it consumes the least energy among the three. Inside the target circle, the thrusters are kept below 10 kN on average, while the angles converge to constant values, see Fig. 20.

Table XV summarizes the performance of the controllers. The MPC has a considerably lower PI compared with the other two controllers, i.e., 0.469e + 03 J×m/s. The model-based controller follows with a PI of 0.691e + 03 J×m/s. The BS controller demonstrates a quite high PI compared with the other two controllers.

Table XVI demonstrates the decrease in the energy consumption when the technique for parasitic thrust reduction is used (see Figs. 4 and 5). For the BS, we observe a considerable decrease of energy because jet angles do not converge to a constant value.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

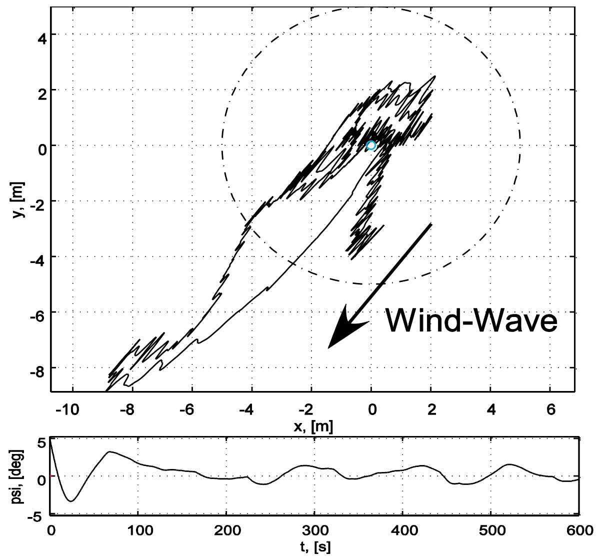

Fig. 19. MB-PID controller dynamic positioning—Task B.

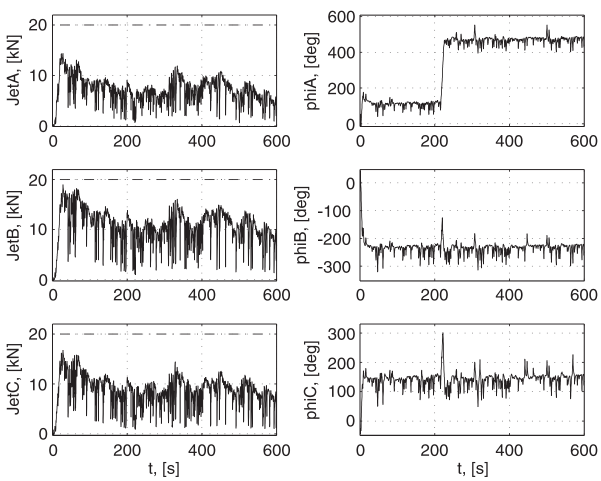

Fig. 20. MB-PID controller jet thrusts/angles—Task B.

We continue with the examination of the robustness for Task B, following the same analysis, as we did for Task A. In Table XVII, robustness results for the MPC are presented. We note that even with a 50% increase in mass, the quantity G is lower than that of the BS and MB-PID with 0% increase in mass. For the MPC, we observe small changes in energy consumption. The MPC PI remains below the BS PI and the MB-PID PI during the entire robustness study process. As a result, the MPC is considered the most robust controller for Task B.

Table XVIII demonstrates the robustness results of the BS controller. The energy consumption is the highest between the three. On the other hand, the G quantity for BS remains below the G for the MB-PID during whole study process.

Table XIX shows the robustness results of the MB-PID controller. The energy consumption is the lowest between the three controllers. However, the accuracy of the dynamic positioning is the worst. The PI of the MB-PID remains below the PI of the BS in the entire study.

TABLE XVII
ROBUSTNESS RESULTS—MPC—TASK B

Increase in mass (%)	0%	10%	30%	50%
E **(J×(e+05))**	1.498	1.510	1.560	1.630
G **(m×s)**	1128	1184	1352	1620
PI **(m× J/s)**	469.3	496.6	585.8	733.5

TABLE XVIII
ROBUSTNESS RESULTS—BS—TASK B

Increase in mass (%)	0%	10%	30%	50%
E **(J×(e+05))**	2.949	3.057	3.111	3.453
G **(m×s)**	1682	1703	1648	1896
PI **(m×J/s)**	1377.8	1446.1	1424.1	1818.5

TABLE XIX
ROBUSTNESS RESULTS—MB-PID—TASK B

Increase in mass (%)	0%	10%	30%	50%
E **(J×(e+05))**	1.355	1.344	1.345	1.359
G **(m×s)**	1839	1836	1882	1975
PI **(m×J/s)**	692.1	685.4	703.1	745.5

TABLE XX
PERFORMANCE FOR TASK B

Controller	***G***	***E***	Robustness
BS	0	-	0
MPC	+	+	+
MB-PID	-	0	0

+ : "very good", 0 : "good", - : " not so good"

2) Conclusion—Task B: MPC has the lowest PI among the three controllers, and it maintains the lowest PI in whole robustness study process. For Task B, we would propose the MPC as the most efficient controller (see Table XV). Table XX summarizes the results for Task B. As a second choice, we would choose the MB-PID, since the PI of the MB-PID remains below the PI of the BS controller. If we focus only on the accuracy of the dynamic positioning, BS wins.

VI. CONCLUSION

We presented the design of a BS and an MPC for an overactuated marine platform. The controllers were evaluated against an MB-PID developed previously. In the design of the controllers, hardware limitations and settling delays in the development of the desired jet thrust and the desired jet angles were taken into consideration. We propose a technique for parasitic thrust reduction, which decreases the energy consumption of the controllers. A comparative study was presented based on simulation results under realistic environmental disturbances. The introduction of the PI was used to reach safer conclusions about which controller accomplishes the dynamic positioning better. The MPC is characterized by the least PI in both tasks and maintains the lowest PI despite errors in platform mass estimates. This is partly due to the augmented model design, which effectively adds an integrator to the controller, resulting in a superior performance of the MPC. The BS exhibits good accuracy in the dynamic positioning, while the MB-PID is our second choice if we are to focus on energy consumption only. The classification of the controllers does not depend on the environmental disturbances. Even if the disturbances are considerably reduced the superiority of the MPC is maintained.

REFERENCES

[1] A. J. Sørensen, "A survey of dynamic positioning control systems," *Annu. Rev. Control*, vol. 35, no. 1, pp. 123–136, Apr. 2011.

[2] T. I. Fossen, *Handbook of Marine Craft Hydrodynamics and Motion Control*, 1st ed. New York, NY, USA: Wiley, 2011.

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

[4] V. Hassani, A. J. Sørensen, and A. M. Pascoal, "A novel methodology for robust dynamic positioning of marine vessels: Theory and experiments," in *Proc. IEEE Amer. Control Conf. (ACC)*, Washington, DC, USA, Jun. 2013, pp. 560–565.

[5] M. G. Feemster and J. M. Esposito, "Comprehensive framework for tracking control and thrust allocation for a highly overactuated autonomous surface vessel," *J. Field Robot.*, vol. 28, no. 1, pp. 80–100, 2011, doi: 10.1002/rob.20369.

[6] L. Xiao and J. Jouffroy, "Modeling and nonlinear heading control of sailing yachts," *IEEE J. Ocean. Eng.*, vol. 39, no. 2, pp. 256–268, Apr. 2014.

[7] Z. Li and J. Sun, "Disturbance compensating model predictive control with application to ship heading control," *IEEE Trans. Control Syst. Technol.*, vol. 20, no. 1, pp. 257–265, Jan. 2012.

[8] T. I. Fossen, K. Y. Pettersen, and R. Galeazzi, "Line-of-sight path following for Dubins paths with adaptive sideslip compensation of drift forces," *IEEE Trans. Control Syst. Technol.*, vol. 23, no. 2, pp. 820–827, Mar. 2015.

[9] N. E. Kahveci and P. A. Ioannou, "Adaptive steering control for uncertain ship dynamics and stability analysis," *Automatica*, vol. 49, no. 3, pp. 685–697, 2013.

[10] A. Witkowska and R. S̀mierzchalski, "Nonlinear backstepping ship course controller," *Int. J. Autom. Comput.*, vol. 6, no. 3, pp. 277–284, 2009.

[11] R. Ghaemi, S. Oh, and J. Sun, "Path following of a model ship using model predictive control with experimental verification," in *Proc. Amer. Control Conf. (ACC)*, Jun./Jul. 2010, pp. 5236–5241.

[12] F. Repoulias and E. Papadopoulos, "Planar trajectory planning and tracking control design for underactuated AUVs," *Ocean Eng.*, vol. 34, nos. 11–12, pp. 1650–1667, Aug. 2007.

[13] L. Medagoda and S. B. Williams, "Model predictive control of an autonomous underwater vehicle in an *in situ* estimated water current profile," in *Proc. OCEANS*, Yeosu, South Korea, May 2012, pp. 1–8.

[14] J. Ghommam, F. Mnif, A. Benali, and N. Derbel, "Asymptotic backstepping stabilization of an underactuated surface vessel," *IEEE Trans. Control Syst. Technol.*, vol. 14, no. 6, pp. 1150–1157, Nov. 2006.

[15] S.-R. Oh and J. Sun, "Path following of underactuated marine surface vessels using line-of-sight based model predictive control," *Ocean Eng.*, vol. 37, nos. 2–3, pp. 289–295, 2010.

[16] S. Mukhopadhyay *et al.*, "Controller performance of marine robots in reminiscent oil surveys," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, Oct. 2012, pp. 1766–1771.

[17] *Nestor Site Characteristics*, accessed on 1992. [Online]. Available: http://www.inp.demokritos.gr/nestor/

[18] F. Repoulias, K. Vlachos, and E. Papadopoulos, "Modeling and control of a triangular floating platform driven by rotating jets," in *Proc. 16th IEEE Medit. Conf. Control Autom. (MED)*, Ajaccio, France, Jun. 2008, pp. 487–492.

[19] K. Vlachos and E. Papadopoulos, "Control design and allocation of an over-actuated triangular floating platform," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2010, pp. 3739–3744.

[20] A. Tsopelakos, K. Vlachos, and E. Papadopoulos, "Backstepping control with energy reduction for an over-actuated marine platform," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, Seattle, WA, USA, May 2015, pp. 553–558.

[21] A. Tsopelakos, K. Vlachos, and E. Papadopoulos, "Design of a linear model predictive controller for an overactuated triangular floating platform," in *Proc. IEEE Conf. Control Appl.*, Antibes, France, Oct. 2014, pp. 891–896.

[22] K. Vlachos and E. Papadopoulos, "Modeling and control of a novel over-actuated marine floating platform," *Ocean Eng.*, vol. 98, pp. 10–22, Apr. 2015.

[23] S. F. Hoerner, *Fluid-Dynamic Drag*. Bakersfield, CA, USA: Hoerner Publications, 1965.

[24] *Recommended Practice for Planning, Designing and Constructing Fixed Offshore Platforms—Working Stress Design*, 21st ed., Amer. Petroleum Inst., Washington, DC, USA, Dec. 2000.

[25] L. Wang, *Model Predictive Control System Design and Implementation Using MATLAB* (Advances in Industrial Control), M. J. Grimble and M. A. Johnson, Eds. London, U.K.: Springer, 2009.

Aristomenis Tsopelakos (S'13) received the Diploma degree in electrical and computer engineering from the National Technical University of Athens, Athens, Greece, in 2014. He is currently pursuing the master's degree with the Coordinated Science Laboratory, Electrical and Computer Engineering Department, University of Illinois at Urbana–Champaign, Champaign, IL, USA.

His current research interests include nonlinear control, predictive control, cybernetics, and robotics.

Evangelos Papadopoulos (S'83–M'91–SM'97) received the Diploma degree from the National Technical University of Athens (NTUA), Athens, Greece, in 1981, and the M.S. and Ph.D. degrees from the Massachusetts Institute of Technology, Cambridge, MA, USA, in 1983 and 1991, respectively, all in mechanical engineering.

He was an Analyst with the Hellenic Navy, Athens, from 1985 to 1987. In 1991, he joined the Centre for Intelligent Machines, McGill University, Montréal, QC, Canada, as an Assistant Professor. He is currently a Professor with the Department of Mechanical Engineering, NTUA. He teaches courses in the areas of systems, controls, mechatronics, and robotics. He has authored over 200 technical articles in journals and refereed conference proceedings. His current research interests include robotics, modeling and control of dynamic systems, mechatronics, and design.

Dr. Papadopoulos is a Senior Member of American Institute of Aeronautics and Astronautics and a member of American Society of Mechanical Engineers (ASME), the Technical Chamber of Greece, and Sigma Xi. He served as an Associate Editor of the IEEE TRANSACTIONS ON ROBOTICS and serves as an Associate Editor of the *ASME Journal of Dynamic Systems, Measurement and Control*, and the *Machine and Mechanism Theory*. He has served as a Guest Editor of the IEEE TRANSACTIONS ON MECHATRONICS/*ASME Transactions on Mechatronics*.