Robust Proportional-Derivative Control on SO(3) for Transporting Quadrotor with Load Uncertainties

Gilang Nugraha Putu Pratama, Adha Imam Cahyadi, and Samiadji Herdjunanto

***Abstract*—In this paper, a model of a transporting quadrotor with proportional-derivative (PD) control on Special Orthogonal-3 is presented. Quadrotor is a unique rotary wing UAV (Unmanned Aerial Vehicle) with special features such as VTOL (Vertical Take-Off and Landing) and hovering at desired altitude. Those features, make it beneficial for many applications such as military purposes, aerial photography, and transporting missions. Unfortunately, due its underactuated nature, while flying it can not be stabilized just from its mechanical structure. Even further, for transporting missions, it is a challenging problem to compensate the inertia perturbation due to the uncertainties of the payload and also input saturation. That is the reason why such an electrical control algorithm needs to be introduced here. Numerical simulation and results are presented to verify the effectiveness of the proposed PD control algorithm.**

***Index Terms*—transporting quadrotor, proportional-derivative control, mathematical model, lyapunov function, special orthogonal-3 group.**

I. Introduction

Quadrotor is a unique rotary-wing UAV, which offers hovering at desired altitude, ease of mechanical structure, high maneuverability, and also take-off and landing vertically. Those features make quadrotor as an excellent platform for many robotic applications [1]–[3]. In recent years, quadrotors are continuously finding new applications in civilian settings such as search and rescue missions, transporting payloads, aerial photography, surveillance, and also for disaster relief operations [4], [5].

As mentioned before, quadrotor has relatively simple mechanical structure. It distinguishes quadrotor with any single-rotor helicopter. By using complex swashplate, single-rotor helicopters are capable to manipulate its attitude. On the other hand, quadrotor manipulates its attitude by controlling the rotational speed of its rotors. Despite its interesting features, quadrotor is a trivially underactuated system. It has 6 degrees of freedom, but with only four actuators. Hence, quadrotor can not be stabilized mechanically. That is the reason why a reliable control algorithm is a must [4]–[7].

Over the years, quadrotor gains more popularity among hobbyists and engineers. There are many kinds of research that have been conducted, including how to represent and control its attitude. The attitude of a rigid body can be represented by a rotation between a reference coordinate frame and a body-fixed coordinate frame. There are several rotational representations of a rigid body such as rotation matrices in Special Orthogonal-3 (SO(3)) group, the Euler angles, and unit quaternions [8].

Arguably, due to its simple approach, the Euler angles representation is the most famous one. It represents the angles, roll (ϕ), pitch (θ), and yaw (ψ), which separately controllable [9]–[11]. Despite its simplicity, with the Euler angles representation being implemented, quadrotor can not avoid rotation matrix singularity. It occurs when roll angle or pitch angle reaches $90°$. This condition leads to a problem which is well known as gimbal lock. Gimbal lock is the loss of one degree of freedom when two of the rotational axes align and locking together. Another drawback of the Euler angles representation is that it neglects the manifold structure of rotation.

On the other hand, both rotation matrices in SO(3) and unit quaternions, offer a singularity-free representation of attitude. However, it is important to be noted that the space of unit quaternions double covers the space of physical attitudes of SO(3). Hence, unit quaternions are not unique [8]. It causes the ambiguity in representing an attitude. This problem should be anticipated cautiously. Otherwise, it turns very sensitive to any small noise, even more, it can yield unwinding behaviors [12], [13].

It has been stated before that quadrotor can be utilized for transporting payloads, even in harsh environment. Practically, it is a challenging problem, since there will be inertia perturbation due to the payload. Therefore, any proposed control algorithm should be able to compensate it. Some notable related works were conducted by [14] and [15]. They proposed adaptive control for transporting quadrotor with the Euler angles representation. Their adaptive control demonstrated a successful result. In spite of that, the adaptive control has drawbacks by nature. It is too sophisticated and consumes relatively high cost for computational. Even further, as the Euler angles representation being implemented the singularity is unavoidable. It constrains the maneuverability of quadrotor.

In real-world experiment, input saturation can occur anytime and sometimes unpredictable. Hence, it is necessary to study motion control with input saturation. Generally speaking, it can be caused by perturbations and the physical constraints of the actuators, as an example the rotor speed [16]. Even further, those physical constraints on states need to be considered for practical reasons [17]. We can take a consideration to constrain its attitude to a safe range to avoid the undesirable configurations.

In this paper, instead of the Euler angles or unit quater-

Manuscript received August 25, 2017; revised March 14, 2018. This work was supported by the Faculty of Engineering, Universitas Gadjah Mada, for providing Publication Research Grant.

Gilang Nugraha Putu Pratama, Adha Imam Cahyadi* (as corresponding author), and Samiadji Herdjunanto, are with the Department of Electrical Engineering and Information Technology, Faculty of Engineering, Universitas Gadjah Mada, Yogyakarta, Indonesia, *email: adha.imam@ugm.ac.id.

nions, the attitude of quadrotor is represented by the exponential coordinates on SO(3) group. Due to its complexity and difficulty to grasp the insight, it becomes less popular than the two others. However, it is important to be noted that the SO(3) group considers rotation in manifold structure. More importantly, it offers features that do not belong to the Euler angles representation and unit quaternions. The SO(3) group does not suffer from any singularity and has unique physical attitudes [8]. Previous works were conducted by [18] and [19]. They prevail to control the attitude on SO(3) group for general purpose quadrotor.

The main objective of this research is to design a model of transporting quadrotor with payload uncertainties and also under input saturation. Due to the large perturbations and initial conditions, or be caused by aggressive upward lift, hence input saturation may occur. It leads the actuator to exceed its capabilities [17]. We need to anticipate the input saturation and consider it carefully. Meanwhile, the payload is described as solid cubic-shaped, where its mass and dimension (depth, height, width) are not known precisely. It causes inertia perturbation which affects the stability of quadrotor. As an inertia perturbation takes place, the proposed PD control algorithm needs to overcome it.

Quadrotor is not just an underactuated system, it is also a second-order system which needs a reliable control algorithm. Technically PD control offers ease of design, simplicity, and reliability for the second-order system. Despite its simplicity, PD controller is capable enough to be applied on quadrotor in order to stabilize the behavior of a model along the desired state [20]–[23]. Numerical simulation and results are presented to verify the effectiveness of the PD control algorithm.

Rest of this paper is organized as follows. The equations of rotation in SO(3) are introduced in section II. There are two issues that will be addressed in section III. The model of quadrotor and its equations are explained first. Later the inertia perturbation and its model, are explained afterward. Section IV explains about the design of the proposed PD control algorithm and its analysis by applying Lyapunov stability theory. Next, numerical simulation, results, and analysis are presented in section V. It verifies the effectiveness of the proposed PD control algorithm. Finally conclusion of this paper is summarized in section VI.

II. Rotation in SO(3) Group

First things first, it is important to explain and distinguish between the $\mathfrak{so}(3)$ and SO(3) group. The term SO(3) stands for Special Orthogonal-3, is a group of 3-by-3 orthogonal matrices. It is called special since its determinant is always +1. Hence, the SO(3) group, can be denoted as

$$\text{SO(3)} = \{\mathbf{R} \in \mathbb{R}^{3\times 3} | \mathbf{R}^{\mathsf{T}}\mathbf{R} = \mathbf{I}_{3\times 3}, \det \mathbf{R} = +1\}, \quad (1)$$

where $\mathbf{R}$ is the rotational matrix and $\mathbf{I}_{3\times 3}$ is a 3-by-3 identity matrix. Meanwhile, the $\mathfrak{so}(3)$ is the Lie algebra of SO(3), which contains skew-symmetric matrices [24]. It can be defined mathematically as

$$\mathfrak{so}(3) = \{\mathbf{S} \in \mathbb{R}^{3\times 3} | \mathbf{S}^{\mathsf{T}} = -\mathbf{S}\}. \quad (2)$$

Later, it is necessary to point out the advantages of this representation. It does not only offer singularity-free representation of attitude, it also has a capability to represent any 3-dimensional rotation of the rigid body with a rotation of a given axis by some amount. This feature distinguishes the exponential coordinates representation from the Euler angles representation. The exponential coordinates representation uses only single arbitrary axis. On the contrary, the Euler angles representation uses composition of three consecutive rotations.

In the same manner with rotation in SO(3) group, the unit quaternions can also represent the attitude of a rigid body without suffering from singularity. Nonetheless, the map space $\mathbb{S}^3$ of unit quaternions to the space of SO(3) is not unique. It means each attitude corresponds to two different quaternion vectors. Precisely, each physical attitude $\mathbf{R} \in$ SO(3) is represented by a pair antipodal of unit quaternions $\pm\mathbf{q} \in \mathbb{S}^3$. More details of this topic can be found in 'Pitfalls of Using Quaternion Representations for Attitude Control' [8].

In this paper, it is assumed that $\boldsymbol{\omega} \in \mathbb{R}^3$ is a unit vector specifying rotation axis and $\vartheta \in \mathbb{R}$ is a rotation angle. Rotation at constant $\boldsymbol{\omega}$ yields the point's translational velocity as

$$\dot{\mathbf{g}}(t) = \boldsymbol{\omega} \times \mathbf{g}(t) = \widehat{\boldsymbol{\omega}}\mathbf{g}(t). \quad (3)$$

The vector $\boldsymbol{\omega} = \begin{bmatrix} \omega_1 & \omega_2 & \omega_3 \end{bmatrix}^{\mathsf{T}}$, is mapped to $\widehat{\boldsymbol{\omega}} \in \mathfrak{so}(3)$, a skew-symmetric matrix, which can be written down as

$$\widehat{\boldsymbol{\omega}} = \begin{bmatrix} 0 & -\omega_3 & \omega_2 \\ \omega_3 & 0 & -\omega_1 \\ -\omega_2 & \omega_1 & 0 \end{bmatrix}. \quad (4)$$

Suppose rotation matrix $\mathbf{R} \in \mathbb{R}^{3\times 3}$, belongs to vector space SO(3) that can be mapped from $\mathfrak{so}(3)$ by applying the exponential operator as

$$\mathbf{R} = \exp(\widehat{\boldsymbol{\omega}}\vartheta) = \mathbf{I}_{3\times 3} + \vartheta\widehat{\boldsymbol{\omega}} + \frac{(\vartheta\widehat{\boldsymbol{\omega}})^2}{2!} + \frac{(\vartheta\widehat{\boldsymbol{\omega}})^3}{3!} + \ldots, \quad (5)$$

which further can be simplified by applying Rodrigues' formula as

$$\mathbf{R} = \mathbf{I}_{3\times 3} + \frac{\widehat{\boldsymbol{\omega}}}{\|\boldsymbol{\omega}\|}\sin(\|\boldsymbol{\omega}\|\vartheta) + \frac{\widehat{\boldsymbol{\omega}}^2}{\|\boldsymbol{\omega}\|^2}(1 - \cos(\|\boldsymbol{\omega}\|\vartheta)). \quad (6)$$

Contrary with equation (5), the logarithmic operator can be utilized for inverse-mapping from SO(3) to $\mathfrak{so}(3)$ as

$$\log(\mathbf{R}) = \frac{\vartheta}{2\sin\vartheta}\left(\mathbf{R} - \mathbf{R}^{\mathsf{T}}\right), \quad (7)$$

where ϑ, by definition of [24], is $\arccos\left(\frac{\text{Tr}(\mathbf{R})-1}{2}\right)$ and $|\vartheta| < \pi$. Actually, there is a condition where we can determine the vector $\boldsymbol{\omega}$ arbitrarily. It holds as long as $\mathbf{R} = \mathbf{I}_{3\times 3}$. Further details can be found in [25]–[27].

III. Modeling

There are two main issues that will be discussed in this section. The model of quadrotor is presented in subsection III-A. Meanwhile in subsection III-B, the inertia perturbation due to the payload is explained.

A. Quadrotor System Model

The rotation matrix $\mathbf{R}$ represents the attitude of quadrotor in exponential coordinates. However, this representation is quite complex, since it has 9 elements. According to [24], it can be simplified by using the angle vector $\boldsymbol{\zeta} = \log(\mathbf{R})^{\vee}$, where $\boldsymbol{\zeta} = \begin{bmatrix} \zeta_1 & \zeta_2 & \zeta_3 \end{bmatrix}^{\mathsf{T}}$.

In the case of attitude control, it is necessary to derive the dynamics of quadrotor in the mathematical model. Still, it is not an easy problem to express the motion of quadrotor with simple modeling. In addition, the inertia perturbation due to the payload has to be modeled. In order to simplify its dynamics model, some assumptions are considered.

- The body and rotors of quadrotor are structured symmetrically and rigid.
- The payload is solid cubic-shaped which contains fluids.
- The payload does not affect the center of the mass of the system.
- There is no external disturbance or uncertainty, but due to the payload and input saturation.

The payload is attached directly to the body of quadrotor. Based on the work of [28], [29], and from Figure 1, the position of quadrotor in the inertial frame is defined by the vector $\mathbf{r} = \begin{bmatrix} x & y & z \end{bmatrix}^{\mathsf{T}}$, and its angular velocity is described in the body frame axes, which can be denoted as $\boldsymbol{\Omega}_b = \begin{bmatrix} p & q & r \end{bmatrix}^{\mathsf{T}}$.

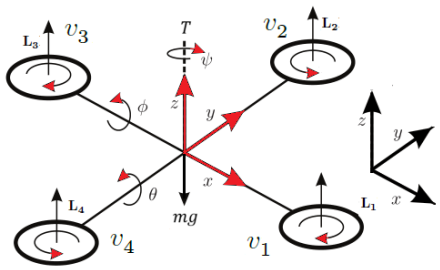

Fig. 1. Quadrotor model.

Here as assumed, that the body of quadrotor is symmetrical and rigid, therefore according to [30], the inertia matrix of quadrotor without the payload can be defined as

$$\mathbf{J_0} = \begin{bmatrix} J_{xx_0} & 0 & 0 \\ 0 & J_{yy_0} & 0 \\ 0 & 0 & J_{zz_0} \end{bmatrix}. \tag{8}$$

Quadrotor's system inputs are the squared angular speeds of its rotors, which can be written down as $\mathbf{u} = \begin{bmatrix} v_1^2 & v_2^2 & v_3^2 & v_4^2 \end{bmatrix}^{\mathsf{T}}$. If the angular speeds of its rotors are regulated properly, then torques can be generated as

$$\boldsymbol{\tau} = \begin{bmatrix} \tau_\phi \\ \tau_\theta \\ \tau_\psi \end{bmatrix} = \begin{bmatrix} lb\left(v_1^2 - v_3^2\right) \\ lb\left(v_2^2 - v_4^2\right) \\ k\left((v_1^2 + v_3^2) - (v_2^2 + v_4^2)\right) \end{bmatrix}, \tag{9}$$

and also for thrust

$$T = b\left(v_1^2 + v_2^2 + v_3^2 + v_4^2\right). \tag{10}$$

Where the constants l, b, and k are respectively the distance from the actuator to the center of the mass, the lift constant, and the aerodynamic drag.

The Euler equation that defines the dynamics of quadrotor is as follows

$$\dot{\boldsymbol{\Omega}}_b = -\mathbf{J_0}^{-1}(\boldsymbol{\Omega}_b \times \mathbf{J_0}\boldsymbol{\Omega}_b) + \mathbf{J_0}^{-1}\boldsymbol{\tau}, \tag{11}$$

and its kinematics can be defined mathematically as

$$\dot{\mathbf{R}} = \mathbf{R}\widehat{\boldsymbol{\Omega}}_b. \tag{12}$$

The differential equation (12) can be represented in terms of the angle vector $\boldsymbol{\zeta} = \log(\mathbf{R})^{\vee}$ as

$$\dot{\boldsymbol{\zeta}} = \left(\mathbf{I}_{3\times 3} + \frac{1}{2}\widehat{\boldsymbol{\zeta}} + [1 - \alpha(\|\boldsymbol{\zeta}\|)]\frac{\widehat{\boldsymbol{\zeta}}^2}{\|\boldsymbol{\zeta}\|^2}\right)\boldsymbol{\Omega}_b, \tag{13}$$

where $\alpha(\|\boldsymbol{\zeta}\|) = \left(\frac{\|\boldsymbol{\zeta}\|}{2}\right)\cot\left(\frac{\|\boldsymbol{\zeta}\|}{2}\right)$. With equation (11) and (13) in mind, the second-order equation of quadrotor on SO(3) can be denoted as

$$\begin{bmatrix} \dot{\boldsymbol{\zeta}} \\ \dot{\boldsymbol{\Omega}}_b \end{bmatrix} = \begin{bmatrix} \left(\mathbf{I}_{3\times 3} + \frac{1}{2}\widehat{\boldsymbol{\zeta}} + [1 - \alpha(\|\boldsymbol{\zeta}\|)]\frac{\widehat{\boldsymbol{\zeta}}^2}{\|\boldsymbol{\zeta}\|^2}\right)\boldsymbol{\Omega}_b \\ -\mathbf{J_0}^{-1}\left(\boldsymbol{\Omega}_b \times \mathbf{J_0}\boldsymbol{\Omega}_b\right) + \mathbf{J_0}^{-1}\boldsymbol{\tau} \end{bmatrix}. \tag{14}$$

Practically speaking, during flight, there will be input saturation, which is due to the large perturbations and initial conditions, or caused by aggressive upward lift. It may lead the actuator to exceed its capabilities. With this fact, the input saturation needs to be considered.

Let the lift $\mathbf{L} = \begin{bmatrix} L_1 & L_2 & L_3 & L_4 \end{bmatrix}^{\mathsf{T}}$, is an upward vector. Unlike in equation (9) and (10), the torques and thrust of quadrotor can be determined in terms of upward lift as

$$\begin{bmatrix} \boldsymbol{\tau} \\ T \end{bmatrix} = \mathbf{M}\begin{bmatrix} L_1 \\ L_2 \\ L_3 \\ L_4 \end{bmatrix}, \tag{15}$$

where $\mathbf{M}$ is a matrix which can be written down as

$$\mathbf{M} = \begin{bmatrix} 0 & l & 0 & -l \\ l & 0 & -l & 0 \\ k & -k & k & -k \\ 1 & 1 & 1 & 1 \end{bmatrix}. \tag{16}$$

The matrix M is of full rank if b, k, and l are positive definite. Hence, the upward lift can be determined as

$$\mathbf{L} = \mathbf{M}^{-1}\begin{bmatrix} \boldsymbol{\tau} \\ T \end{bmatrix}, \tag{17}$$

where

$$\mathbf{M}^{-1} = \begin{bmatrix} 0 & \frac{1}{2l} & \frac{1}{4k} & \frac{1}{4} \\ \frac{1}{2l} & 0 & -\frac{1}{4k} & \frac{1}{4} \\ 0 & -\frac{1}{2l} & \frac{1}{4k} & \frac{1}{4} \\ -\frac{1}{2l} & 0 & -\frac{1}{4k} & \frac{1}{4} \end{bmatrix}. \tag{18}$$

Currently, how the input saturation will perturb the proposed PD control algorithm, is still obscure. It will be explained in the next section. At this point, it is obvious that the upward lift revises the dynamics of quadrotor as in equation (11).

B. Inertia Perturbation

Previously, it has been assumed that the inertia matrix is undisturbed. There is no perturbation in the inertia matrix. Now let taking the perturbation in inertia matrix due to the payload, $\Delta\mathbf{J}$, into consideration. At this point, only the nominal inertia matrix, $\mathbf{J}_0$, is determined. Meanwhile, the value of perturbed inertia matrix, $\mathbf{J}$, is still unknown. The relation between them can be expressed as

$$\mathbf{J} = \mathbf{J_0} + \Delta\mathbf{J}, \tag{19}$$

where $\Delta\mathbf{J}$ is defined as

$$\Delta\mathbf{J} = \begin{bmatrix} \Delta J_{xx} & \Delta J_{xy} & \Delta J_{xz} \\ \Delta J_{yx} & \Delta J_{yy} & \Delta J_{yz} \\ \Delta J_{zx} & \Delta J_{zy} & \Delta J_{zz} \end{bmatrix}. \tag{20}$$

Consider it as a symmetrical matrix, therefore $\Delta J_{xy} = \Delta J_{yx}$, $\Delta J_{xz} = \Delta J_{zx}$, and $\Delta J_{yz} = \Delta J_{zy}$. The model of the payload is cubic-shaped as shown in Figure 2. It contains fluids, as assumed in subsection III-A.

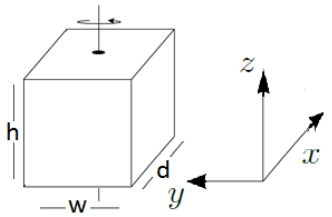

Fig. 2. Model of the payload that will be transported.

Therefore, the diagonal elements of inertia matrix perturbation that has been defined before in equation (20) can be derived as

$$\begin{aligned} \Delta J_{xx} &= \frac{m}{12}(d^2 + h^2), \\ \Delta J_{yy} &= \frac{m}{12}(d^2 + w^2), \\ \Delta J_{zz} &= \frac{m}{12}(h^2 + w^2), \end{aligned} \tag{21}$$

where m, h, d, and w respectively are the total mass, height, depth, and width of the payload. Those parameters are generated randomly. Even so, the parameters of the payload are constrained at certain intervals. The total mass of the load is constrained within 0.5 to $1\,\mathrm{kg}$. The empty mass of the payload is $0.01\,\mathrm{kg}$ and the mass of contained fluids is the rest. Simultaneously, it is within 0.05 to $0.30\,\mathrm{m}$, for the rest of parameters. This approach is conducted since in the real implementation, sometimes the parameters of the payload, that will be transported, are not known precisely.

By all means, things are getting more complicated, since small changes in its parameters will affect the inertia perturbation drastically.

IV. PD Control

The system of quadrotor described by equation (14), can be made asymptotically stable by the following control algorithm [24],

$$\boldsymbol{\tau} = \boldsymbol{\Omega}_b \times \mathbf{J_0}\boldsymbol{\Omega}_b - \mathbf{K}_p\boldsymbol{\zeta} - \mathbf{K}_d\boldsymbol{\Omega}_b, \tag{22}$$

where $\mathbf{K}_p$ and $\mathbf{K}_d$ are positive definite matrices. It is apparent now, how the upward lift causes problems to the proposed PD control algorithm. As in the attitude control of quadrotor, the torques take significance role. The attitude of quadrotor is determined by the torques, in which will alter due to the upward lift.

The closed-loop system satisfies

$$\begin{aligned} \dot{\boldsymbol{\zeta}} &= \sum_{n=0}^{\infty} \frac{(-1)^n \mathrm{B}_n}{n!} \mathrm{ad}_{\boldsymbol{\zeta}}^n (\boldsymbol{\Omega}_b) = \beta_{\boldsymbol{\zeta}}\boldsymbol{\Omega}_b, \\ \dot{\boldsymbol{\Omega}}_b &= -\mathbf{J_0}^{-1}\mathbf{K}_p\boldsymbol{\zeta} - \mathbf{J_0}^{-1}\mathbf{K}_d\boldsymbol{\Omega}_b, \end{aligned} \tag{23}$$

where B_n are Bernoulli numbers, for further explanation can be seen at [24].

Such a nonlinear system can be determined asymptotically stable in the sense of Lyapunov as long as V is positive definite and its time derivative, $\dot{V}$, is negative definite [31]–[33]. Consider a Lyapunov function candidate $V(x) : \mathbb{R}^n \to \mathbb{R}$ such that

$$\begin{aligned} V(x) = 0 &\iff x = 0, \\ V(x) > 0 &\iff x \neq 0, \\ \dot{V}(x) &= \frac{d}{dt}V(x) \\ &= \sum_{i=1}^{n} \frac{\partial V}{\partial x_i} f_i(x), \end{aligned} \tag{24}$$

where $\dot{V}(x) \leq 0, \forall x \neq 0$, which is negative semidefinite, hence the system is stable. Meanwhile, for asymptotic stability, it needs to be negative definite, $\dot{V}(x) < 0, \forall x \neq 0$.

The following Lyapunov function can be utilized to prove the stability of quadrotor under control algorithm (22)

$$V = \frac{1}{2}\left\langle \begin{bmatrix} \boldsymbol{\zeta} \\ \boldsymbol{\Omega}_b \end{bmatrix}, \begin{bmatrix} \mathbf{I}_{3\times3} & \epsilon\mathbf{I}_{3\times3} \\ \epsilon\mathbf{I}_{3\times3} & \mathbf{K}_p^{-1}\mathbf{J_0} \end{bmatrix} \begin{bmatrix} \boldsymbol{\zeta} \\ \boldsymbol{\Omega}_b \end{bmatrix} \right\rangle, \tag{25}$$

where $\mathbf{I}_{3\times3}$ is a 3-by-3 identity matrix. It can be simplified as

$$V = \frac{1}{2}\boldsymbol{\xi}^{\mathsf{T}}\mathbf{P}_{\epsilon}\boldsymbol{\xi}, \tag{26}$$

where

$$\boldsymbol{\xi} = \begin{bmatrix} \boldsymbol{\zeta} \\ \boldsymbol{\Omega}_b \end{bmatrix}, \tag{27}$$

and

$$\mathbf{P}_{\epsilon} = \begin{bmatrix} \mathbf{I}_{3\times3} & \epsilon\mathbf{I}_{3\times3} \\ \epsilon\mathbf{I}_{3\times3} & \mathbf{K}_p^{-1}\mathbf{J_0} \end{bmatrix}. \tag{28}$$

The time derivative of Lyapunov function (25) is

$$\begin{aligned} \dot{V} &= \left\langle \boldsymbol{\zeta}, \dot{\boldsymbol{\zeta}} \right\rangle + \left\langle \boldsymbol{\Omega}_b, \mathbf{K}_p^{-1}\mathbf{J_0}\dot{\boldsymbol{\Omega}}_b \right\rangle \\ &\quad + \epsilon \left\langle \beta_{\boldsymbol{\zeta}}\boldsymbol{\Omega}_b, \boldsymbol{\Omega}_b \right\rangle + \epsilon \left\langle \boldsymbol{\zeta}, \dot{\boldsymbol{\Omega}}_b \right\rangle \\ &= \left\langle \boldsymbol{\zeta}, \boldsymbol{\Omega}_b \right\rangle + \left\langle \boldsymbol{\Omega}_b, \mathbf{K}_p^{-1}\mathbf{J_0}\dot{\boldsymbol{\Omega}}_b \right\rangle \\ &\quad + \epsilon \left\langle \beta_{\boldsymbol{\zeta}}\boldsymbol{\Omega}_b, \boldsymbol{\Omega}_b \right\rangle + \epsilon \left\langle \boldsymbol{\zeta}, \dot{\boldsymbol{\Omega}}_b \right\rangle \\ &= \left\langle \boldsymbol{\zeta}, \boldsymbol{\Omega}_b \right\rangle + \left\langle \boldsymbol{\Omega}_b, \mathbf{K}_p^{-1}\mathbf{J_0}\left(-\mathbf{J_0}^{-1}\mathbf{K}_p\boldsymbol{\zeta} - \mathbf{J_0}^{-1}\mathbf{K}_d\boldsymbol{\Omega}_b\right) \right\rangle \\ &\quad + \epsilon \left\langle \beta_{\boldsymbol{\zeta}}\boldsymbol{\Omega}_b, \boldsymbol{\Omega}_b \right\rangle + \epsilon \left\langle \boldsymbol{\zeta}, -\mathbf{J_0}^{-1}\mathbf{K}_p\boldsymbol{\zeta} - \mathbf{J_0}^{-1}\mathbf{K}_d\boldsymbol{\Omega}_b \right\rangle \\ &\leq -\left\langle \boldsymbol{\Omega}_b, \mathbf{K}_p^{-1}\mathbf{K}_d\boldsymbol{\Omega}_b \right\rangle + \epsilon \left\langle \boldsymbol{\Omega}_b, \boldsymbol{\Omega}_b \right\rangle \\ &\quad - \epsilon \left\langle \boldsymbol{\zeta}, \mathbf{J_0}^{-1}\mathbf{K}_p\boldsymbol{\zeta} \right\rangle - \epsilon \left\langle \boldsymbol{\zeta}, \mathbf{J_0}^{-1}\mathbf{K}_d\boldsymbol{\Omega}_b \right\rangle. \end{aligned} \tag{29}$$

The simplification of the first term in the first line of equation (29) uses the fact that

$$\left\langle \boldsymbol{\zeta}, \dot{\boldsymbol{\zeta}} \right\rangle = \left\langle \boldsymbol{\zeta}, \boldsymbol{\Omega}_b \right\rangle, \tag{30}$$

as used in [24]. In addition, the following upper bound stated in [24] is used

$$\left\langle \beta_{\boldsymbol{\zeta}}\boldsymbol{\Omega}_b, \boldsymbol{\Omega}_b \right\rangle \leq \left\langle \boldsymbol{\Omega}_b, \boldsymbol{\Omega}_b \right\rangle. \tag{31}$$

Next, the inequality (29) can be simplified as

$$\dot{V} \leq -\boldsymbol{\xi}^{\mathsf{T}}\mathbf{Q}_{\epsilon}\boldsymbol{\xi}, \tag{32}$$

where

$$\mathbf{Q}_{\epsilon} = \begin{bmatrix} \epsilon\mathbf{J_0}^{-1}\mathbf{K}_p & \frac{\epsilon}{2}\mathbf{J_0}^{-1}\mathbf{K}_d \\ \frac{\epsilon}{2}\mathbf{J_0}^{-1}\mathbf{K}_d & \mathbf{K}_p^{-1}\mathbf{K}_d - \epsilon\mathbf{I}_{3\times3} \end{bmatrix}. \tag{33}$$

The matrix $\mathbf{Q}_{\epsilon}$ is positive definite for very small ϵ.

From equation (26) and (32), in order V to be positive definite and $\dot{V}$ to be negative definite, $\mathbf{P}_\epsilon$ and $\mathbf{Q}_\epsilon$ need to be made positive definite. The $\mathbf{P}_\epsilon$ and $\mathbf{Q}_\epsilon$ can be defined as positive definite matrices, as long as the conditions are satisfied. Firstly, the ϵ needs to be a small positive value near zero, let $\epsilon = 0.0001$. The second condition is the gains, $\mathbf{K}_p$ and $\mathbf{K}_d$, have to be positive definite matrices. Hence it will satisfy conditions for positive definite matrix by leading principal minor.

V. Numerical Simulation and Analysis

There are five main scenarios that will be simulated. The first scenario is conducted to determine the gains, $\mathbf{K}_p$ and $\mathbf{K}_d$, in nominal condition without any perturbation. In the first scenario, the proposed PD control algorithm needs to stabilize the attitude of quadrotor to its origin. On the other hand, for the second scenario and further, the proposed PD algorithm has to regulate the attitude of quadrotor to its desired condition. The second scenario is conducted under dedicated initial conditions while being under inertia perturbation. The third one, we decide to vary several initial conditions for $\boldsymbol{\zeta}(0)$, with constant inertia matrix perturbation due to the payload. Afterward, in the fourth scenario time-varying inertia perturbation due to the payload is introduced. Last but not least, the transporting quadrotor under input saturation is presented in the fifth scenario. The proposed PD algorithm has to stabilize the attitude of quadrotor, regardless the presence of inertia perturbation due to the payload and input saturation.

Before any further, it is important to determine the parameters of the system. The model of quadrotor that will be used in this simulation is based on [34] and [35] with

$$\begin{aligned} J_{xx_0} &= 0.082\,\mathrm{kg\,m^2}, \\ J_{yy_0} &= 0.0845\,\mathrm{kg\,m^2}, \\ J_{zz_0} &= 0.1377\,\mathrm{kg\,m^2}. \end{aligned} \tag{34}$$

In real-world experiment, the maneuver of quadrotor needs to be quick, responsive, and smooth. If there is any perturbation, the control algorithm should counter it as swift as possible. Therefore, to confirm its effectiveness and efficiency, we decide that the simulation will be conducted just in 5 seconds for every scenario.

The model of quadrotor that will be used in this simulation is based on [34] and [35], and for the first and second scenario, the initial conditions of the system are $\boldsymbol{\zeta}(0) = \begin{bmatrix} 29 & -8 & -29 \end{bmatrix}^\mathsf{T}$ and $\boldsymbol{\Omega}_b(0) = 0$. The initial conditions are in degrees, since it is more convenient rather than in radians.

A. First Scenario: Nominal Condition

The first scenario of simulation is conducted under nominal condition. It means the inertia perturbation due to the payload has not been introduced yet. Here we decide to conduct four experiments to determine the most suitable gains for $\mathbf{K}_p$ and $\mathbf{K}_d$. Those gains are shown in Table I.

The gains, $\mathbf{K}_p$ and $\mathbf{K}_d$, are positive definite matrices, just as requirement mentioned before in the last part of section IV. The results are shown in Figure 3, 4, and 5. It demonstrates that in nominal condition, the system is quickly convergent. It also verifies that the proposed PD control algorithm can make the system asymptotically stable for this condition. The $\boldsymbol{\zeta}(0)$ can recover from $\begin{bmatrix} 29 & -8 & -29 \end{bmatrix}^\mathsf{T}$ to the origin. Here the term origin refers to the desired set point $\boldsymbol{\zeta}_d = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}^\mathsf{T}$.

TABLE I
Gains ($\mathbf{K}_p$ and $\mathbf{K}_d$) for each experiment

First Scenario		
Experiment	$\mathbf{K}_p$	$\mathbf{K}_d$
Experiment 1	$10\mathbf{I}_{3\times3}$	$4\mathbf{I}_{3\times3}$
Experiment 2	$12\mathbf{I}_{3\times3}$	$8\mathbf{I}_{3\times3}$
Experiment 3	$12\mathbf{I}_{3\times3}$	$4\mathbf{I}_{3\times3}$
Experiment 4	$10\mathbf{I}_{3\times3}$	$8\mathbf{I}_{3\times3}$

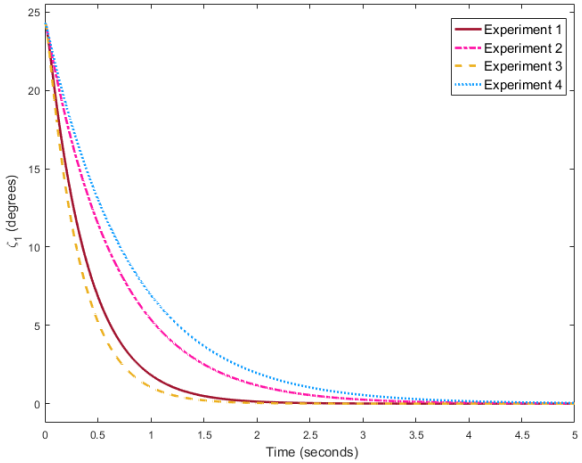

Fig. 3. The angle ζ_1 in nominal condition with several gains.

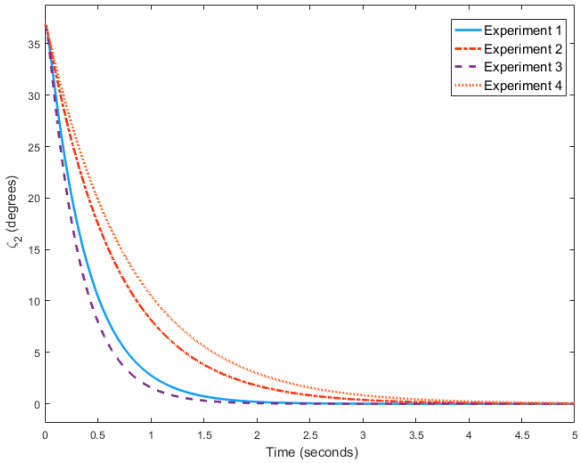

Fig. 4. The angle ζ_2 in nominal condition with several gains.

It is apparent from Figure 3, 4, and 5, that the gains, $\mathbf{K}_p = 12\mathbf{I}_{3\times3}$ and $\mathbf{K}_d = 4\mathbf{I}_{3\times3}$, has better rise time and settling time than the others. It convergences to the origin just about 2 seconds. Hence for the next scenario, it is determined that $\mathbf{K}_p = 12\mathbf{I}_{3\times3}$ and $\mathbf{K}_d = 4\mathbf{I}_{3\times3}$.

B. Second Scenario: Under Inertia Perturbation

The second scenario is conducted under influence of diagonal inertia matrix perturbation. The off-diagonal elements should be 0. Meanwhile, the diagonal elements are

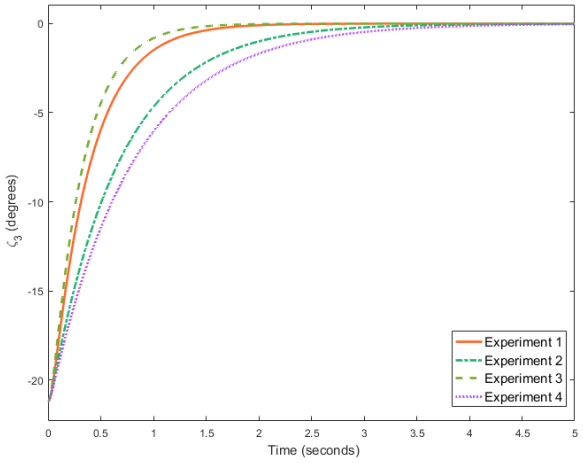

Fig. 5. The angle ζ_3 in nominal condition with several gains.

constrained within certain values. It has been stated before in subsection III-B, that any changes in the parameters of the payload will affect the value of its diagonal elements, ΔJ_{xx}, ΔJ_{yy}, and ΔJ_{zz}.

Unlike in the first scenario, from here and further, the proposed PD algorithm has to track the desired condition $\boldsymbol{\zeta}_d = \begin{bmatrix} 60 & 90 & -60 \end{bmatrix}^{\mathsf{T}}$. In this scenario the initial condition is $\boldsymbol{\zeta}(0) = \begin{bmatrix} 29 & -8 & -29 \end{bmatrix}^{\mathsf{T}}$.

The inertia perturbation due to the payload is represented in the term of Frobenius norm, $\|\Delta\mathbf{J}\|_{\mathbf{F}}$. It is also called Hilbert–Schmidt norm. The norm itself can be considered as a vector norm which is also equal to the square root of the matrix trace of $\Delta\mathbf{J}\Delta\mathbf{J}^{\mathsf{H}}$, where $\Delta\mathbf{J}^{\mathsf{H}}$ is the conjugate transpose, hence

$$\|\Delta\mathbf{J}\|_{\mathbf{F}} = \sqrt{\mathrm{Tr}\,(\Delta\mathbf{J}\Delta\mathbf{J}^{\mathsf{H}})}. \tag{35}$$

The $\|\Delta\mathbf{J}\|_{\mathbf{F}}$ can also be defined as the square root of the sum of the absolute squares of its elements. It can be written down mathematically as

$$\|\Delta\mathbf{J}\|_{\mathbf{F}} = \sqrt{\sum_{i=1}^{3}\sum_{j=1}^{3}|J_{ij}|^2}, \tag{36}$$

where $\Delta\mathbf{J}$ is a 3×3 matrix.

Table II shows various values of $\|\Delta\mathbf{J}\|_{\mathbf{F}}$ for five different experiments.

TABLE II
THE INERTIA PERTURBATIONS

Second Scenario	
Experiment	$\|\Delta\mathbf{J}\|_{\mathbf{F}}$
Experiment 1	0.0147
Experiment 2	0.0106
Experiment 3	0.0259
Experiment 4	0.0113
Experiment 5	0.0128

The largest possible value of Frobenius norm, $\|\Delta\mathbf{J}\|_{\mathbf{F}}$ is 0.0259. It is determined by the upper bounds of the parameters of the payload. The angles, ζ_1, ζ_2, and ζ_3, can recover from $\boldsymbol{\zeta}(0)$ to reach the desired condition $\boldsymbol{\zeta}_d$ just about in 2 seconds. It is obvious that regardless the inertia perturbation, the angles are almost perfectly collided, as shown in Figure 7, 8, and 9. It indicates that the proposed PD control algorithm prevails to regulate the angle vector $\boldsymbol{\zeta}$ to its desired condition $\boldsymbol{\zeta}_d$. Furthermore we can analyze the angle vector $\boldsymbol{\zeta}$ in 3-D plot as shown in Figure 6.

C. Third Scenario: Varying Initial Conditions

In the third scenario, by varying the initial condition for $\boldsymbol{\zeta}(0)$, the proposed PD control algorithm needs to regulate it to reach the desired condition $\boldsymbol{\zeta}_d = \begin{bmatrix} 60 & 90 & -60 \end{bmatrix}^{\mathsf{T}}$. In addition, quadrotor is under constant inertia perturbation $\|\Delta\mathbf{J}\|_{\mathbf{F}}$ due to the payload.

The second scenario indicates that as long as the values of the parameters do not exceed the upper bounds in subsection III-B, hence the $\|\Delta\mathbf{J}\|_{\mathbf{F}}$ does not affect the stability of the system drastically. In this scenario, we have $\|\Delta\mathbf{J}\|_{\mathbf{F}} = 0.0259$, as the value of the inertia perturbation. There are five experiments of initial conditions for $\boldsymbol{\zeta}(0)$, as presented in Table III.

TABLE III
INITIAL CONDITIONS FOR $\boldsymbol{\zeta}(0)$

Third Scenario			
Experiment	ζ_1 (degrees)	ζ_2 (degrees)	ζ_3 (degrees)
Experiment 1	-22	-4	-31
Experiment 2	58	53	-58
Experiment 3	38	-14	-38
Experiment 4	26	68	-18
Experiment 5	-1	41	25

The details for the angles ζ_1, ζ_2, and ζ_3 are shown in Figure 11, 12, and 13. The angles converge to the desired condition over the time.

The angles, ζ_1, ζ_2, and ζ_3, converge at the desired condition $\boldsymbol{\zeta}(0)$ about 3 seconds.

D. Fourth Scenario: Time-Varying Inertia Perturbation

This scenario is inspired by [36]. Here, things are getting more complicated. In the previous work [36], they propose a model and attitude control for quadrotor under variations of mass. Their model has a constant sploshing rate. On the other hand, in this paper the sploshing rate is a random value. It also changes over the time.

The initial parameters of the payload is presented, as shown in Table IV.

TABLE IV
INITIAL PARAMETERS OF THE PAYLOAD

Fourth Scenario		
Parameter	Value	Unit
Initial mass	1	kg
Height	0.3	m
Width	0.3	m
Depth	0.3	m

The sploshing rate is clearly presented as in Figure 14.

Since the total mass of the payload is changing over the time, it implies that the inertia perturbation due to the payload is also varying, as shown in Figure 15.

Up until now, the proposed PD control algorithm prevails to overcome such a constant inertia perturbation due to the

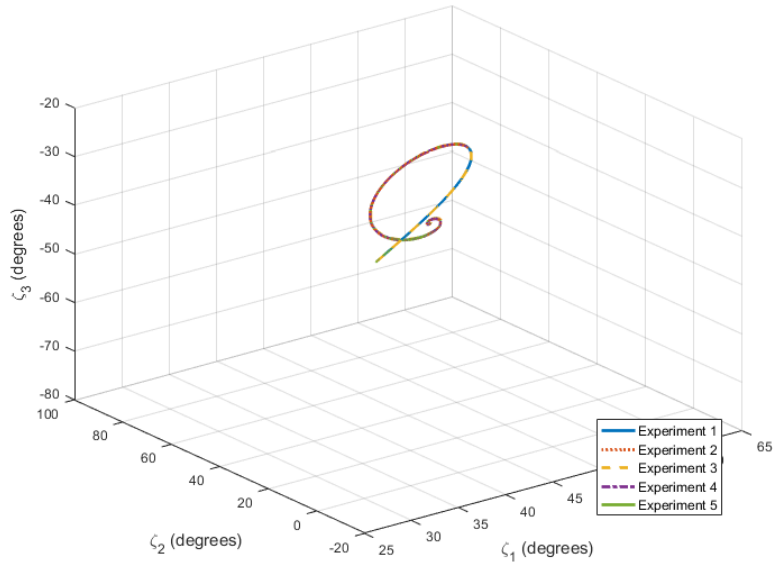

Fig. 6. The angle vector ζ in 3-D plot under inertia perturbation.

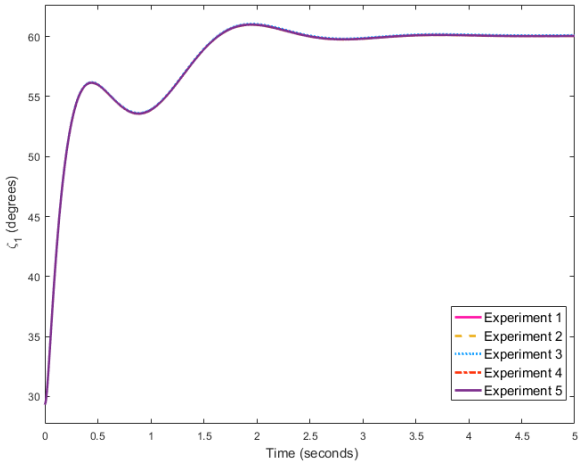

Fig. 7. The angle ζ_1 under inertia perturbation.

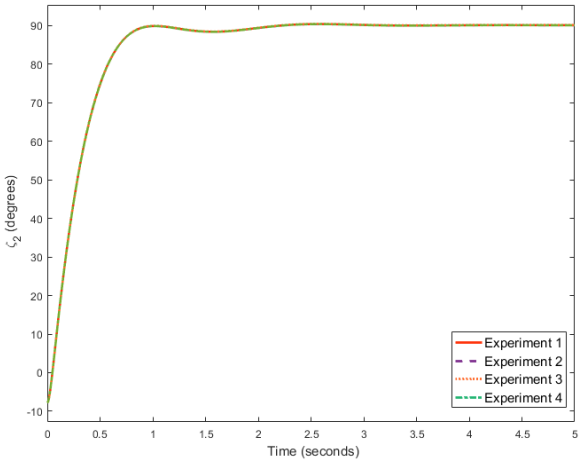

Fig. 8. The angle ζ_2 under inertia perturbation.

payload. At present, the proposed PD algorithm needs to deal with time-varying inertia perturbation.

Despite under time-varying inertia perturbation due to the payload, the proposed PD control algorithm successfully to regulate the attitude of quadrotor at desired state

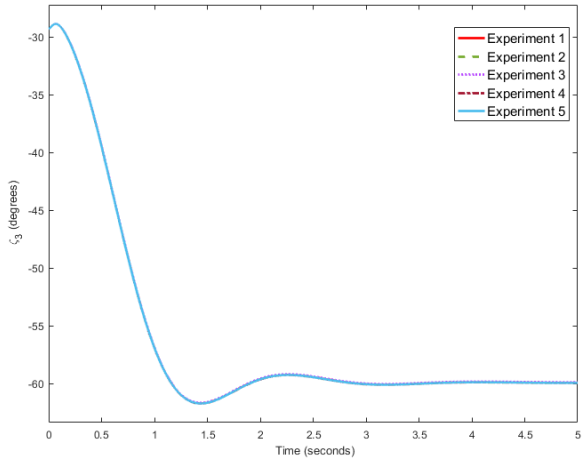

Fig. 9. The angle ζ_3 under inertia perturbation.

as shown in Figure 16. It reaches the desired condition $\zeta_d = \begin{bmatrix} 60 & 90 & -60 \end{bmatrix}^\mathsf{T}$ from the initial condition $\zeta(0) = \begin{bmatrix} 29 & -8 & -29 \end{bmatrix}^\mathsf{T}$, just about 2 seconds.

E. Fifth Scenario: Transporting Under Input Saturation

Unlike in the previous scenarios, in the last one, the input saturation due to the upward lift is considered. Firstly, we need to determine the parameters in equation (16), $l = 0.165\,\mathrm{m}$, the distance from the actuator to the center of the mass, and $k = 0.016$, the aerodynamic drag. In this scenario, the desired condition is not merely $\zeta_d = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}^\mathsf{T}$, we also introduce the desired condition for the thrust as

$$T_d = (L_1 + L_2 + L_3 + L_4) = 4. \tag{37}$$

The rest of parameters, the payload and initial conditions, are same as in the fourth scenario. Whereas, the value of the upward lift can vary from $0\,\mathrm{N}$ up to $10\,\mathrm{N}$. Now we will look to how the proposed PD control algorithm deals with input saturation as shown in Figure 17 and 18.

We can see clearly that the values of each parameter in Figure 17 and 18 do not deviate anymore after 2 seconds. At

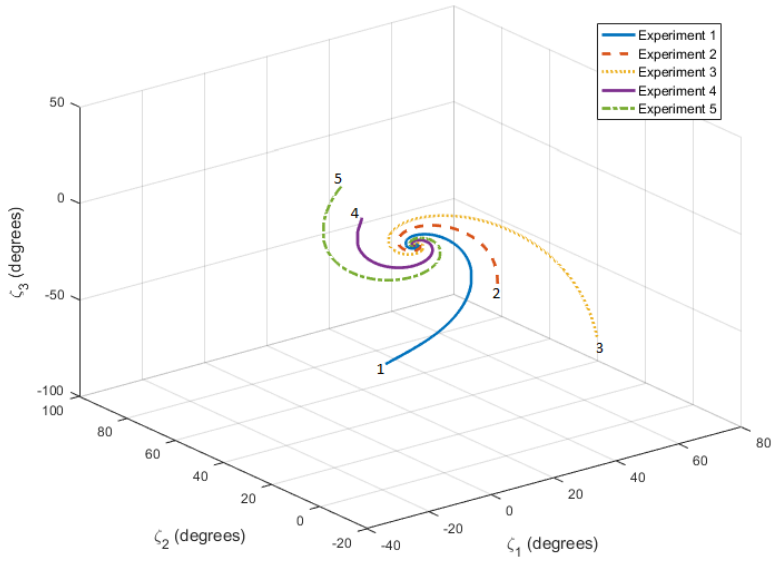

Fig. 10. The experiments of initial conditions $\zeta(0)$ under inertia perturbation in 3-D plot.

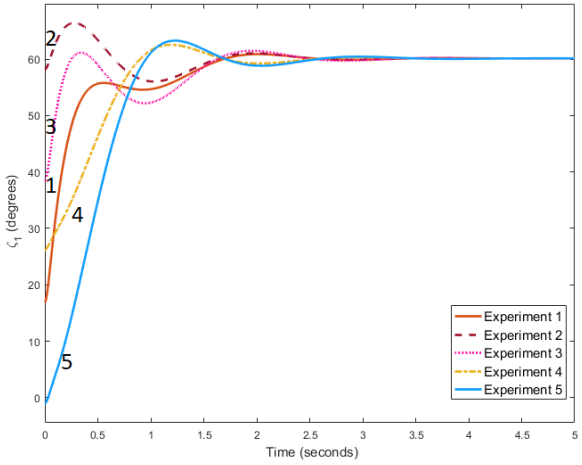

Fig. 11. The experiments of initial conditions ζ_1 under inertia perturbation.

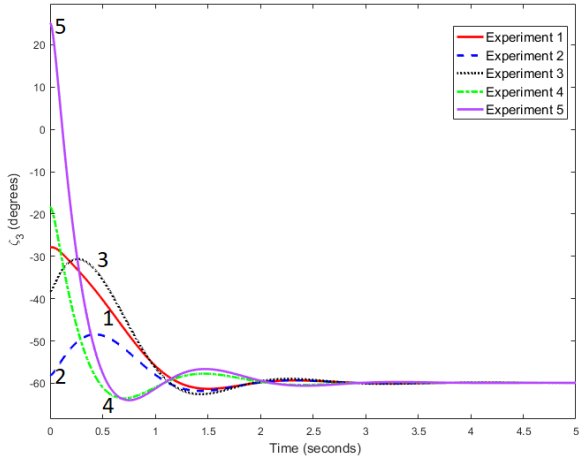

Fig. 13. The experiments of initial conditions ζ_3 under inertia perturbation.

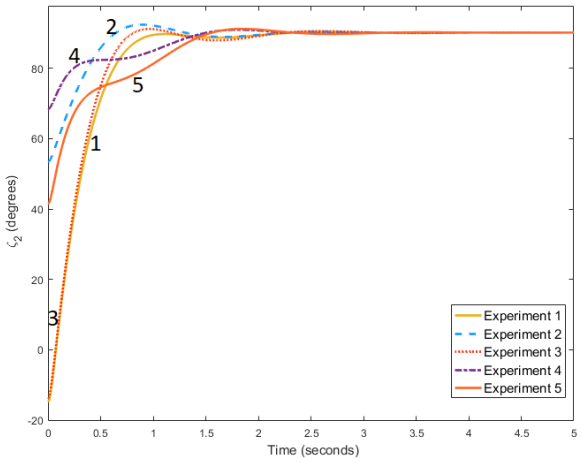

Fig. 12. The experiments of initial conditions ζ_2 under inertia perturbation.

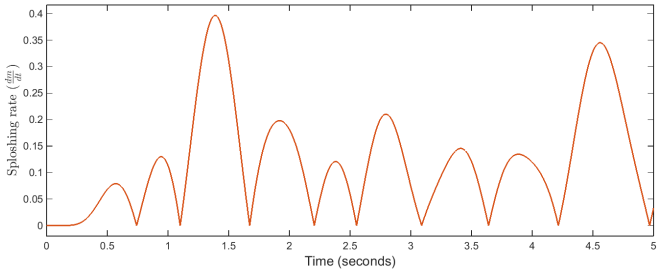

Fig. 14. The sploshing rate.

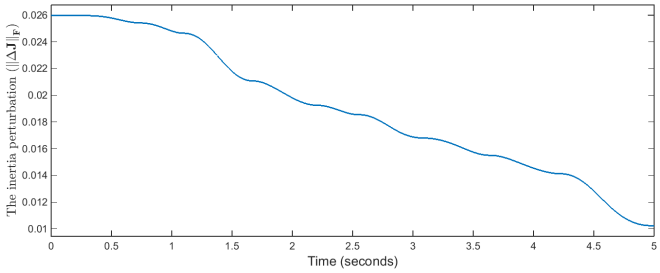

Fig. 15. The inertia perturbation with sploshing rate.

the beginning of the simulation, the pair opposites ($L1$ with $L2$ and $L3$ with $L4$) of the upward lifts are in the extreme values. It yields the angles, ζ_1, ζ_2, and ζ_3, deviate from the desired condition.

Regardless the presence of input saturation, the proposed

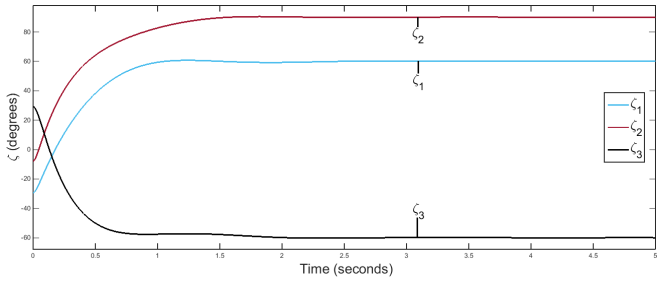

Fig. 16. The angle vector ζ under time-varying inertia perturbation.

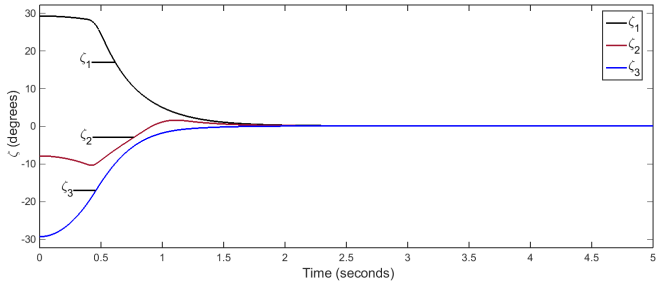

Fig. 17. The angle vector ζ under input saturation.

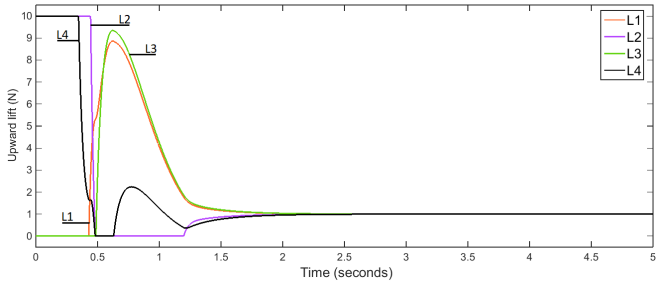

Fig. 18. The upward lift respectively to each actuator of quadrotor.

PD control algorithm succeeds to regulate the attitude of the transporting quadrotor to its desired condition. The desired condition for $\zeta_d = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}^\mathsf{T}$ can be fulfilled as long as the pair opposites of the upward lifts are equal. It implies that $L_1 - L_2 + L_3 - L_4 = 0$. Now, let turning back to equation (37), we have $\sum_{n=1}^{4} L_n = 4\,\mathrm{N}$. Those conditions can be satisfied if $L_1 = L_2 = L_3 = L_4 = 1\,\mathrm{N}$ as expressed in Figure 18 at steady state condition. Those values converge just about in 2 seconds.

VI. Conclusion

In this paper, model of quadrotor and its attitude control on SO(3) are presented. Simulation and numerical analysis provide a verification of the effectiveness of the proposed PD control algorithm. Based on the overall results, it can be stated that the proposed PD control algorithm is capable to handle perturbation due to the payload and also the effect of input saturation. It is worth to be noted that despite under time-varying inertia perturbation and initial condition the proposed PD control algorithm prevails to regulate the attitude of quadrotor to its desired condition with relatively quick responses.

Acknowledgment

We gratefully appreciate the Faculty of Engineering, Universitas Gadjah Mada, for providing Publication Research Grant (Penyiapan Publikasi Internasional) to support this research.

References

[1] A. Gorbenko, M. Mornev, and V. Popov, "Planning a typical working day for indoor service robots," *IAENG International Journal of Computer Science*, vol. 38, no. 3, pp. 176–182, 2011.

[2] R. Mahony, V. Kumar, and P. Corke, "Multirotor aerial vehicles: Modeling, estimation, and control of quadrotor," *IEEE Robotics Automation Magazine*, vol. 19, no. 3, pp. 20–32, Sept 2012.

[3] H. Lim, J. Park, D. Lee, and H. J. Kim, "Build your own quadrotor: Open-source projects on unmanned aerial vehicles," *IEEE Robotics Automation Magazine*, vol. 19, no. 3, pp. 33–45, Sept 2012.

[4] F. Mohammed, A. Idries, N. Mohamed, J. Al-Jaroodi, and I. Jawhar, "Uavs for smart cities: Opportunities and challenges," in *2014 International Conference on Unmanned Aircraft Systems (ICUAS)*, May 2014, pp. 267–273.

[5] O. A. Dhewa, A. Dharmawan, and T. K. Priyambodo, "Model of linear quadratic regulator (lqr) control method in hovering state of quadrotor," *Journal of Telecommunication, Electronic and Computer Engineering*, vol. 9, no. 3, pp. 135 – 143, 2017.

[6] I. C. Dikmen, A. Arisoy, and H. Temeltas, "Attitude control of a quadrotor," in *2009 4th International Conference on Recent Advances in Space Technologies*, June 2009, pp. 722–727.

[7] A. Dharmawan, A. Ashari, A. E. Putra, and A. Harjoko, "The use of video processing for quadrotor flight stability control monitoring," *AIP Conference Proceedings*, vol. 1755, no. 1, p. 170008, 2016.

[8] N. A. Chaturvedi, A. K. Sanyal, and N. H. McClamroch, "Rigid-body attitude control," *IEEE Control Systems*, vol. 31, no. 3, pp. 30–51, June 2011.

[9] M. Sghairi, A. de Bonneval, Y. Crouzet, J.-J. Aubert, and P. Brot, "Challenges in building fault -tolerant flight control system for a civil aircraft," *IAENG International Journal of Computer Science*, vol. 35, no. 4, pp. 495–499, 2008.

[10] Z. Zhong and D. Pi, "Forecasting satellite attitude volatility using support vector regression with particle swarm optimization," *IAENG International Journal of Computer Science*, vol. 41, no. 3, pp. 153–162, 2014.

[11] A. Dharmawan, A. Ashari, and A. E. Putra, "Quadrotor flight stability system with routh stability and lyapunov analysis," *AIP Conference Proceedings*, vol. 1755, no. 1, p. 170007, 2016.

[12] T. Lee, "Exponential stability of an attitude tracking control system on so(3) for large-angle rotational maneuvers," *Systems & Control Letters*, vol. 61, no. 1, pp. 231 – 237, 2012.

[13] T.-H. Wu and T. Lee, "Angular velocity observer for attitude tracking on so(3) with the separation property," *International Journal of Control, Automation and Systems*, vol. 14, no. 5, pp. 1289–1298, Oct 2016.

[14] B. C. Min, J. H. Hong, and E. T. Matson, "Adaptive robust control (arc) for an altitude control of a quadrotor type uav carrying an unknown payloads," in *2011 11th International Conference on Control, Automation and Systems*, Oct 2011, pp. 1147–1151.

[15] Z. Wang, S. Liang, T. Liu, and L. Zhang, "Robust adaptive attitude control of quadrotors with load uncertainties," in *2016 Chinese Control and Decision Conference (CCDC)*, May 2016, pp. 5962–5967.

[16] F. Mazenc, R. E. Mahony, and R. Lozano, "Forwarding control of scale model autonomous helicopter: a lyapunov control design," in *42nd IEEE International Conference on Decision and Control (IEEE Cat. No.03CH37475)*, vol. 4, Dec 2003, pp. 3960–3965 vol.4.

[17] M. Bürger and M. Guay, "A backstepping approach to multivariable robust constraint satisfaction with application to a vtol helicopter," in *Proceedings of the 48h IEEE Conference on Decision and Control (CDC) held jointly with 2009 28th Chinese Control Conference*, Dec 2009, pp. 5239–5244.

[18] T. Fernando, J. Chandiramani, T. Lee, and H. Gutierrez, "Robust adaptive geometric tracking controls on so(3) with an application to the attitude dynamics of a quadrotor uav," in *2011 50th IEEE Conference on Decision and Control and European Control Conference*, Dec 2011, pp. 7380–7385.

[19] Y. Yu, S. Yang, M. Wang, C. Li, and Z. Li, "High performance full attitude control of a quadrotor on so(3)," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*, May 2015, pp. 1698–1703.

[20] B. Erginer and E. Altug, "Modeling and pd control of a quadrotor vtol vehicle," in *2007 IEEE Intelligent Vehicles Symposium*, June 2007, pp. 894–899.

[21] H. Liu, Y. Bai, G. Lu, and Y. Zhong, "Brief paper - robust attitude control of uncertain quadrotors," *IET Control Theory Applications*, vol. 7, no. 11, pp. 1583–1589, July 2013.

[22] M. Walid, N. Slaheddine, A. Mohamed, and B. Lamjed, "Modeling and control of a quadrotor uav," in *2014 15th International Conference on Sciences and Techniques of Automatic Control and Computer Engineering (STA)*, Dec 2014, pp. 343–348.
[23] S. Bencharef and H. Boubertakh, "Optimal tuning of a pd control by bat algorithm to stabilize a quadrotor," in *2016 8th International Conference on Modelling, Identification and Control (ICMIC)*, Nov 2016, pp. 938–942.
[24] F. Bullo and R. M. Murray, "Proportional derivative (pd) control on the euclidean group," in *3rd European Control Conference*, 1995, pp. 1091–1097.
[25] R. M. Murray, S. S. Sastry, and L. Zexiang, *A Mathematical Introduction to Robotic Manipulation*, 1st ed. Boca Raton, FL, USA: CRC Press, Inc., 1994.
[26] S. Herdjunanto, A. P. Sandiwan, and A. I. Cahyadi, "Quadrotor proportional-derivative regulation for nonzero set point on so(3) with disturbance compensation," *IOP Conference Series: Materials Science and Engineering*, vol. 190, no. 1, p. 012005, 2017.
[27] A. P. Sandiwan, A. Cahyadi, and S. Herdjunanto, "Robust proportional-derivative control on so(3) with disturbance compensation for quadrotor uav," *International Journal of Control, Automation and Systems*, vol. 15, no. 5, pp. 2329–2342, Oct 2017.
[28] M. T. Hussein and M. N. Nemah, "Modeling and control of quadrotor systems," in *2015 3rd RSI International Conference on Robotics and Mechatronics (ICROM)*, Oct 2015, pp. 725–730.
[29] A. Dharmawan, A. Ashari, and A. E. Putra, "Mathematical modelling of translation and rotation movement in quad tiltrotor," *International Journal on Advanced Science, Engineering and Information Technology*, vol. 7, no. 3, pp. 1104–1113, 2017.
[30] D. Morin, *Introduction to classical mechanics: with problems and solutions*. Cambridge University Press, 2008.
[31] H. K. Khalil, *Nonlinear Systems*, 3rd ed. Prentice Hall, 2002.
[32] A. N. Al-Rabadi, "Intelligent control of singularly-perturbed reduced order eigenvalue-preserved quantum computing systems via artificial neural identification and linear matrix inequality transformation," *IAENG International Journal of Computer Science*, vol. 37, no. 3, pp. 210–223, 2010.
[33] M. H. Mabrouk, "Crowd behavior simulation using artificial potential fields," *IAENG International Journal of Computer Science*, vol. 40, no. 4, pp. 220–229, 2013.
[34] P. Pounds, R. Mahony, and P. Corke, "Modelling and control of a large quadrotor robot," *Control Engineering Practice*, vol. 18, no. 7, pp. 692–699, July 2010.
[35] P. Corke, *Robotics, Vision and Control Fundamental Algorithms in MATLAB*. Springer-Verlag Berlin Heidelberg, 2011.
[36] S. Lee, D. K. Giri, and H. Son, "Modeling and control of quadrotor uav subject to variations in center of gravity and mass," in *2017 14th International Conference on Ubiquitous Robots and Ambient Intelligence (URAI)*, June 2017, pp. 85–90.

Gilang Nugraha Putu Pratama is currently student at Department of Electrical Engineering and Information Technology, Universitas Gadjah Mada, Indonesia. His research interest involves robotics and intelligence systems.

Adha Imam Cahyadi obtained his bachelor degree from Department of Electrical Engineering, Faculty of Engineering, Universitas Gadjah Mada in 2002. Later he got his master in Control Engineering from KMITL in 2005, Thailand, and Doctor of Engineering from Tokai University Japan in 2008. He is curently lecturer in Department of Electrical Engineering and Information Technology, Universitas Gadjah Mada. His research areas involves mechanical control systems, telemanipulation systems, and Unmanned Aerial Vehicles.

Samiadji Herdjunanto obtained his bachelor degree from Department of Electrical Engineering, Faculty of Engineering, Universitas Gadjah Mada in 1979. Later he got his master degree from Ohio State University, USA in 1985, and doctoral degree from Universitas Gadjah Mada in 2015. He is curently lecturer in Department of Electrical Engineering and Information Technology, Universitas Gadjah Mada. His research areas involves control, signal processing, fault detection, isolation and reconstruction.