Hardware-in-the-loop emulation of the optimal adaptive model-free control algorithm for autonomous mobile robots

Ali Safaei

ARTICLE HISTORY
Compiled March 6, 2019

ABSTRACT
In this paper, the performance of recently proposed optimal adaptive model-free control (OAMFC) algorithm for autonomous mobile robots is evaluated in a hardware-in-the-loop (HIL) test platform. The algorithm is a solution for tracking problem in completely unknown nonlinear dynamic systems operated under unknown external disturbances. Here, the OAMFC algorithm alongside with a standard Kalman-filter for eliminating the measurement noise is embedded on a microcontroller and the dynamic model of the autonomous mobile robot is implemented on another microcontroller. The data between two microcontollers is transferred using an I^2C communication protocol. In addition, the data is logged during the HIL test in MATLAB using serial communication between the microcontrollers and a laptop. Appropriate performance of the OAMFC algorithm is observed based on the results for the emulated dynamic models of a quadrotor and a wheeled mobile robot (WMR).

1. Introduction

Autonomous mobile robots are among the most attractive robotic products with several applications in industry, agriculture, warehouses and surveillance projects. Their applications are spreading rapidly in almost all of the industries from the automated mass production line to the transportation of goods and also to the automated advisory inspections over the final products. The main goal is to provide fully-autonomous functionality for the mobile robots performing autonomous tasks such as manufacturing, transportation and inspection projects. The dynamics of mobile robots are naturally nonlinear and has several unknown terms. Diverse environmental condition exerts various perturbations on the mobile robot dynamics, which can be regarded as unknown disturbances. Hence,the need for a control algorithm which can be adapted online without requiring complete dynamic information of the system and the ambient environment is very essential. Several investigations have been proposed for addressing this need. Model-free control Fliess et al. (2013); Percup et al. (2017) and adaptive reinforcement learning algorithms Vamvoudakis et al. (2014); Zhu et al. (2016) are among the most famous and well-developed adaptive algorithms to provide fully autonomous functionality. Recently, the authors have proposed the optimal adaptive model-free control (OAMFC) algorithm as a solution for tracking problem in completely unknown dynamic systems Safaei et al. (2018,-). The algorithm includes two model-free parameter estimators which exempt the persistently excitation (PE) condition for any associated regressors. This implies that the sufficiently rich (SR) condition for input reference can be waived as a result. In addition, the main controller gain is updated on-

line using an adaptive form of a differential Riccati equation. The computer simulation results for implementing the OAMFC algorithm on a robotic manipulator, a vibration resonator and also mobile robots are presented in Safaei et al. (2018-2, 2017,-, 2018-3), respectively. Moreover, a detailed comparison study for the OAMFC algorithm against the state-of-the-art model-free and reinforcement learning algorithms is presented in Safaei et al. (2018,-).

HIL test platform is a useful step implementing the algorithm on the real product for experimental validation. Due to this advantage, HIL test platforms have wide range of applications in aircraft and missile industries Qi et al. (2017), automotive industry Verma et al. (2008); Michalek et al. (2015) and motion control of robotics White et al. (2009); Li et al. (2016); Cheron et al. (2010). In the cases that the dynamic model of the plant to be controlled is not available easily or it is too expensive to provide several numbers of the real plant for testing and trial efforts, the HIL test platform would be helpful for validating the control algorithm Bayrakceken et al. (2011); Khan et al. (2014). Besides, HIL test serves as a safe validation method for a crucial control application where the system platform is delicate or potentially cause hazard to life being. Moreover, the appropriate ranges for tuning parameters of the control algorithm can be found during several tests using HIL test platform.

On the way toward the real-life application of the OAMFC algorithm, the HIL tests for implementing the algorithm on autonomous mobile robots are presented in this paper. Here, the HIL test platform includes two development boards (Arduino Mega 2560 boards) which are communicating with each other using the I^2C communication protocol. The boards are connected to a laptop using USB cables for real-time data logging. In the following, first the OAMFC algorithm is introduced in Section II. The proofs of the algorithm can be found in Safaei et al. (2018-2). Then, the HIL test platform is proposed with more details in Section III. This section also includes the mathematical models for emulating the dynamics of the mobile robots, as well as the formulation of the state-observer for removing the measurement noise on the states of the robots. The state observer for the translational motion is a Kalman-filter, while a well-known complimentary filter is used for the angular motion of the mobile robots. Finally, the results logged during the HIL test of a quadrotor and a WMR are presented in Section IV. An analysis on the effects of the values for amplitudes of the external disturbance and the measurement noise, is also included in this section.

2. OAMFC algorithm

Definition 1. Assume that the unknown dynamic system of a generic MIMO nonlinear plant with n states and m control input variables subjected to unknown disturbances, can be proposed as follows Safaei et al. (2018-2)

$$\dot{x} = f(x, u, t) \ , \tag{1}$$

where $x \in \mathbb{R}^{n\times 1}$ is the vector of system's states and $u \in \mathbb{R}^{m\times 1}$ is the vector of control inputs, which includes virtual control variables alongside the real control variables Safaei et al. (2018-2). The virtual control variables are mostly defined in second-order dynamic systems to provide access to the system's states that cannot be tuned directly using the real control variables Khalil et al. (2002). In (1), t is the parameter to show the time. Here, $f(.) \in \mathbb{R}^{n\times 1}$ is a vector including the unknown nonlinear functions and unknown external disturbances.

Definition 2. The nonlinearities $f(.)$ in the dynamic system in (1) can be considered to have an unknown linear-in-states part, a linear-in-control variables term and a separate unknown time-varying nonlinear function which is state-dependent Safaei et al. (2018,-). Based on this consideration, the dynamic system in (1) is represented as Safaei et al. (2018-2)

$$\dot{x} = Ax + Bu + g(x,t) \ , \tag{2}$$

where $A \in \mathbb{R}^{n\times n}$ is a diagonal matrix including the unknown linear-in-states parameters and $g(.) \in \mathbb{R}^{n\times 1}$ is a vector of unknown nonlinear functions depending only on system's states and time (not the control inputs). It should be noted that, A is considered to be a diagonal matrix and includes only the values for direct dependencies of the dynamics of each state to that state, itself. All of the coupling terms among the states are assumed to be included in the unknown term g. As it is shown in our recent results for a robotic manipulator Safaei et al. (2017), the coupling dynamics of system's states can be supposed to be included in $g(.)$, without any degradation in performance of the OAMFC algorithm (Fig. (1)). Here, $B \in \mathbb{R}^{n\times m}$ is an invertible input matrix which includes zero and one values, indicating the dependencies between each system state and each of the control variables.

Definition 3. For a tracking problem, the desired trajectory to be followed by the dynamic system is defined as $x_d \in \mathbb{R}^{n\times 1}$. Then, a tracking error can be defined as Safaei et al. (2018-2)

$$\sigma = e + \zeta \ , \tag{3}$$

where

$$e = x_d - x \ , \tag{4}$$

and $\zeta = \int e dt$. The tracking objective is to guarantee that $\sigma \to 0$ as $t \to \infty$.

Proposition 1. For a generic dynamic system in (2), the unknown terms (i.e. A and g) can be adapted on-line using the OAMFC algorithm via the following estimators Safaei et al. (2018-2)

$$\begin{aligned} \dot{\hat{g}} &= -\Gamma_1 P\sigma - \rho_1 \Gamma_1 \hat{g} \\ \dot{v}_{\hat{A}} &= -\Gamma_2 P \mathcal{M}_\sigma (x-\sigma) - \rho_2 \Gamma_2 v_{\hat{A}} \ , \end{aligned} \tag{5}$$

where $\hat{g}$ is the estimated values for unknown g, $v_{\hat{A}} \in \mathbb{R}^{n\times 1}$ is a vector including the diagonal elements of $\hat{A}$ (which is a diagonal matrix as the estimated matrix for unknown A), $\mathcal{M}_\sigma \in \mathbb{R}^{n\times n}$ is a diagonal matrix whose elements on the main diameter are the elements of the vector σ. The diagonal matrices Γ_1 and Γ_2 in $\mathbb{R}^{n\times n}$ are the constant adaptation gains for estimating the elements of $\hat{g}$ and $v_{\hat{A}}$, respectively. The ρ_1 and ρ_2 are two small constant leakage gains Safaei et al. (2018-2). Here, the positive definite matrix $P \in \mathbb{R}^{n\times n}$ is defined using the following differential Riccati equation

(DRE) Safaei et al. (2018-2)

$$\dot{P} = \hat{A}^T P + P\hat{A} - PBRB^T P + 2Q \ , \tag{6}$$

where, $R \in \mathbb{R}^{m\times m}$ and $Q \in \mathbb{R}^{n\times n}$ are positive definite constant matrices.
Proof. Detailed proof can be found in Safaei et al. (2018-2).

Proposition 2. In order to satisfy the tracking objective defined in *Definition 3*, the control inputs in the OAMFC algorithm are determined as $u = u_1 + u_2$, where Safaei et al. (2018-2)

$$\begin{aligned} u_1 &= \frac{1}{2} RB^T P\sigma \\ u_2 &= B^{-1}[\dot{x}_d - \hat{A}x - \hat{g} - \zeta + (I_n + 2P^{-1}Q + \hat{A})\sigma] \\ &\quad - \frac{3}{4} RB^T P\sigma \ . \end{aligned} \tag{7}$$

Here, the values of P, $\hat{g}$ and $\hat{A}$ are defined referring to *Proposition 1*. In addition, I_n is an identity matrix in $\mathbb{R}^{n\times n}$. The values for $\dot{x}_d$ are defined using the sliding-mode differentiators as follows Safaei et al. (2018-2)

$$\dot{x}_d = \eta \ , \tag{8}$$

where for $x_d = [x_{di}]^T$ and $i = \{1, 2, ..., n\}$, we have

$$\begin{aligned} \dot{w}_i &= \eta_i \\ \eta_i &= -k_1 |w_i - x_{di}|^{\frac{1}{2}} sgn(w_i - x_{di}) + \tau_i \\ \dot{\tau}_i &= -k_2 sgn(w_i - x_{di}) \ , \end{aligned} \tag{9}$$

where $w = [w_i]^T$, $\eta = [\eta_i]^T$ and $\tau = [\tau_i]^T$ are three vectors in $\mathbb{R}^{n\times 1}$ and k_1 and k_2 are two positive constants. Moreover, it is shown in Safaei et al. (2018,-) that the term u_1 in (7), satisfies the Hamilton-Jacobi-Bellman optimality equation.
Proof. Detailed proof can be found in Safaei et al. (2018-2).

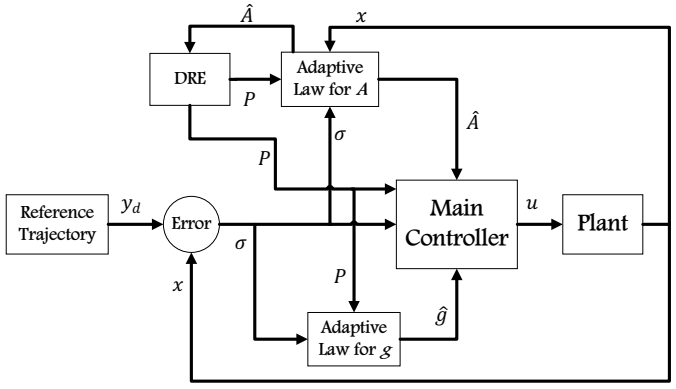

Figure 1. The flowchart of OAMFC algorithm Safaei et al. (2018-2)

3. HIL test platform

The HIL test platform constitutes two development boards. The OAMFC algorithm is implemented on one of the boards (board-A), while the other board (board-B) is emulating the dynamics of a real mobile robot (either WMR or quadrotor) based on a proven dynamic model. The dynamic models for a WMR and a quadrotor are presented in the following subsections. The scheme of the HIL test platform is depicted in Fig. (2). Here, the code implementing the OAMFC algorithm in board-A is optimized in the way to have the least possible computation cost for the algorithm. The forward Euler integration is employed for numerical computation. The communication between the board-A and board-B is provided using the I^2C protocol. Here, the speed of I^2C communication is set to 400 KB/s and the maximum buffer size for the communicated message in each single loop of the algorithm is 64 byte. Moreover, the baud rate of the serial communication for logging data in MATLAB is set to 115 200 KB/s, in order to ensure fast data logging process and thereby the sampling time for the algorithm and the emulated dynamic model can be conveniently lowered. The achieved sampling time for the HIL test is about 5 millisecond.

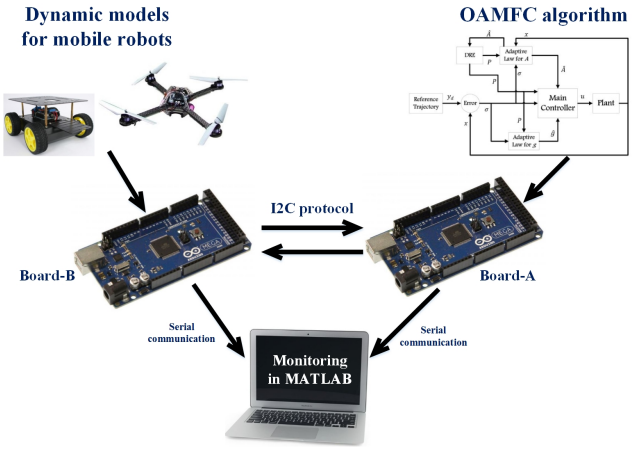

Figure 2. The HIL test platform for real-time evaluation of the OAMFC algorithm on mobile robots.

3.1. Dynamic model for a WMR

The mathematical model for states dynamics of a WMR is considered as follows Safaei et al. (2018-2)

$$\begin{aligned} \dot{x}_w &= v_w \cos(\theta_w) \quad , \quad \dot{y}_w = v_w \sin(\theta_w) \quad , \quad \dot{\theta}_w = w_w \\ \dot{v}_w &= \frac{1}{M_w}[F_w - K_F v_w + f_w] \\ \dot{w}_w &= \frac{1}{J_w}[\tau_w - K_\tau w_w + t_w] \ . \end{aligned} \tag{10}$$

The position of WMR in plane and its heading are defined as $[x_w, y_w, \theta_w]$. In addition, the linear and angular velocities of the WMR are presented by v_w and w_w, respectively. The parameters M_w and J_w are mass and inertia. K_F and K_τ are the coefficients for

frictional force and frictional torque, respectively. Moreover, the unknown disturbances on the WMR are presented by f_w and t_w, respectively. Besides, the total force and torque generated by the set of two electric motors located on the two sides of the WMR are computed as follows Safaei et al. (2018-2)

$$\begin{aligned} F_w &= \frac{k_e}{R_w}(w_{1w} + w_{2w}) \\ \tau_w &= \frac{k_e L_w}{R_w}(w_{1w} - w_{2w}) \ , \end{aligned} \tag{11}$$

where R_w is the wheel radius, L_w is the half of distance between the wheels and k_e is the torque constant of each electric motor. The speed of electric motors (i.e. w_{1w} and w_{2w}) are the control inputs to the WMR model. The values for parameters in the WMR model are defined as $M_w = 1$, $J_w = 0.1$, $K_F = K_\tau = k_e = 0.1$, $R_w = L_w = 0.1$, $f_w = \alpha_w \sin(0.1t)$ and $t_w = 0.1 \times \alpha_w \sin(0.1t)$, where parameter t is representing time and $\alpha_w \in \mathbb{R}$ is the constant amplitude of the external disturbance acting on the WMR.

3.2. Dynamic model for a quadrotor

The dynamic model of quadrotor is as follows Safaei et al. (2018-2)

$$\begin{aligned} \dot{\vec{p}_q} &= \vec{v_q} \quad , \quad \dot{\vec{\Phi}}_q = R_{qt}^{-1} \vec{w_q} \\ \dot{\vec{v}_q} &= \frac{1}{M_q}[R_q \vec{F_q} - K_d \vec{v_q} - M_q \vec{F_g} + R_q \vec{f_q}] \\ \dot{\vec{w}_q} &= J_q^{-1}[\vec{\tau_q} - K_a \vec{w_q} - \vec{w_q} \times J_q \vec{w_q} + \vec{t_q}] \ , \end{aligned} \tag{12}$$

where $\vec{p_q} = [x_q; y_q; z_q]$ and $\vec{\Phi}_q = [\phi_q; \theta_q; \psi_q]$ are the absolute positions and Euler angles (roll, pitch and yaw) of the quadrotor, respectively. In addition, $\vec{v_q}$ and $\vec{w_q}$ are the vectors of linear and angular velocities. The parameter M_q is the quadrotor mass and $J_q \in \mathbb{R}^{3\times3}$ is its inertia matrix. The parameters K_d and K_a are two constants coefficients for the friction forces and torques acting against the movement of the quadrotor. The vector $\vec{F_g} = [0; 0; g]$ is the vector of gravity force and $\vec{f_q}$ and $\vec{t_q}$ are the vectors of unknown disturbances. Here, the body frame to inertia frame transformation is defined by two matrices R_q and R_{qt} Safaei et al. (2018-2). The generated force and torques by the set of four electric motors on the quadrotor are computed by Safaei et al. (2018-2)

$$\begin{bmatrix} F_T \\ \tau_x \\ \tau_y \\ \tau_z \end{bmatrix} = \begin{bmatrix} k_l & k_l & k_l & k_l \\ -k_l L_q & 0 & k_l L_q & 0 \\ 0 & -k_l L_q & 0 & k_l L_q \\ k_t & -k_t & k_t & -k_t \end{bmatrix} \begin{bmatrix} w_{1q}^2 \\ w_{2q}^2 \\ w_{3q}^2 \\ w_{4q}^2 \end{bmatrix} \ , \tag{13}$$

where $\vec{F_q} = [0; 0; F_T]$ and $\vec{\tau_q} = [\tau_x; \tau_y; \tau_x]$. Here, k_l and k_t are the constants of electric motors for generating lift force and torque, respectively. The parameter L_q is the length for each arm of the quadrotor and w_{iq} for $i = \{1, 2, 3, 4\}$ are the angular speeds of the electric motors. The values of these parameters in the model are $L_q = 0.25\ m$, $M_q = 1.95\ Kg$, $J_q = 1.24 \times 10^{-3} \times diag(1, 1, 2)\ kg.m^2$, $K_d = K_a = 10^{-3}$, $k_l = 10^{-5}$, $k_t = 4.35 \times 10^{-7}$, $g = 9.81\ m/s^2$, $\vec{f_q} = \sin 0.3t\ \vec{v}_1\ N$ and $\vec{t_q} = \sin 0.3t\ \vec{v}_3\ N.m$, where

$\vec{v}_1 = [0; 0; \alpha_q]$ and $\vec{v}_3 = \alpha_q \times [1; 1; 1]$, where $\alpha_q \in \mathbb{R}$ is the constant amplitude of the external disturbance acting on the quadrotor.

3.3. OAMFC implementation

Proposition 3. For implementing the OAMFC algorithm on the WMR dynamics, we have used two separate OAMFC units corresponding to the linear and rotational motions of the WMR. The idea comes to deal with the non-holonomic characteristic of the WMR dynamics. In fact, it is assumed that the WMR first adjust its heading toward the goal position, then it tunes its speed in order to reach the destination toward its fixed heading. Based on this assumption, there should be two consecutive OAMFC units for regulating the heading and the linear speed. In this case, each of the OAMFC units has two system states and two control inputs. The angular displacement θ_w and the rotational speed w_w of the WMR are the states in rotational motion. On the other hand, the projected linear displacement p_w as Safaei et al. (2018-2)

$$p_w = x_w \cos(\theta_w) + y_w \sin(\theta_w), \tag{14}$$

and the linear velocity v_w of the WMR are the states in linear motion. The real control input in rotational motion is the generated moment τ_w, while the generated force F_w is the real control input for linear motion of the WMR. Moreover, there is one virtual control input in each of the aforementioned motions for the WMR. The values for actual control inputs w_{1w} and w_{2w} can be determined using the set of equations in (11).

Proposition 4. For implementation of the OAMFC algorithm on a quadrotor, there are 12 system states and 12 control inputs. System states are the linear displacements $\vec{p}_q$, rotational displacements $\vec{\Phi}_q$ and their corresponding rates of changes, i.e. $\vec{v}_q$ and $\vec{w}_q$. In addition, here we have 12 control inputs, including 6 real and 6 virtual control variables. The components of the generated force in the body frame of the quadrotor (i.e. $F_q^{Body} = R_q \vec{F}_q$) are the 3 real control inputs corresponding to the linear motion, where the value F_T is determined as Safaei et al. (2018-2)

$$F_T = \sqrt{(F_q^{Body}(1))^2 + (F_q^{Body}(2))^2 + (F_q^{Body}(3))^2}\ . \tag{15}$$

Besides, the generated torques $\vec{\tau}_q$ are the 3 real control variables related to the angular motion of the quadrotor. Hence, the actual control variables for the quadrotor can be computed using the set of equations presented in (13).

Proposition 5. It should be declared that all of the parameters in the model of WMR are considered to be unknown for design and implementation of the OAMFC algorithm (including k_e, L_w and R_w in (11)). In the case of quadrotor, the values for all of the parameters are unknown (including k_t and L_q in (13)) except the value of k_l which is the constant for computing the lift force of each propeller based on its rotational speed. This value can be determined in an initial test (measuring the angular speed of electric motors, while the quadrotor is hovering) prior to the real application of the OAMFC on a quadrotor.

3.4. State observer for mobile robots

The OAMFC algorithm can be used only for handling the unknown disturbances and the unknown internal dynamics of a mobile robot (including aerial, ground or underwater mobile robot). In order to eliminate the random measurement noise of the measured data received from the on-board sensors, we should use state observers or filters. Here, we have divided the states into two parts as the first part includes the linear displacement and the linear velocity, while the second one constitutes the angular displacement and its rate of change.

3.4.1. Linear motion

For part one, a simple Kalman-filter based on the linear kinematics of the mobile robots is used for observing the states Cui et al. (2016). It is assumed that the absolute or local linear displacement of the mobile robot can be measured using GPS or UWB sensors and there is not any on-board sensor for measuring the local or absolute linear velocity of the mobile robot. On the other hand, there is an Inertia Measurement Unit (IMU) sensor available for measuring the linear acceleration. In this regard, the general kinematics of the linear motion for a mobile robot can be considered as follows Cui et al. (2016)

$$\begin{aligned} \dot{\hat{x}} &= A_o\hat{x} + B_o u_o + G_o w_o \\ z_o &= H_o\hat{x} + v_o \ , \end{aligned} \tag{16}$$

where $\hat{x}_o \in \mathbb{R}^{n\times 1}$ is the observed system states of linear motion (including the linear displacement and the linear velocity), $u_o \in \mathbb{R}^{n\times 1}$ is the measured dynamics of system states (the linear acceleration) and $z_o \in \mathbb{R}^{p\times 1}$ is the measured states of the system. Here, p is the number of available measurements among the total system states. As it is mentioned, here only the data for linear displacement is available. In (16), $A_o \in \mathbb{R}^{n\times n}$ is the known state matrix for the system kinematics, $B_o \in \mathbb{R}^{n\times r}$ is the dynamics measurement matrix, $G_o \in \mathbb{R}^{n\times n}$ is the input matrix for the noise on dynamic measurements and $H_o \in \mathbb{R}^{p\times n}$ is the measurement matrix. These matrices are known and depending on the linear kinematics of the mobile robot. The specific values of the variables in (16) for the case of mobile robots are presented in Table 1. In this table, $\mathbf{0}_3$ is a matrix in $\mathbb{R}^{3\times 3}$ with all elements are equal to zero, while $\mathbf{I}_3 \in \mathbb{R}^{3\times 3}$ and $\mathbf{I}_6 \in \mathbb{R}^{6\times 6}$ are identity matrices. In addition, the variables with superscript m are the measured variables using the on-board sensors. Note that w_o and v_o are vectors of random noise, exist in the state and output equations, respectively. The first equation in (16) can be considered as the process model for implementing the standard Kalman-filter, while the second equation in (16) is the measurement model. Taking this into account, a continuous Kalman-filter is used as follows Lewis et al. (2007)

$$\begin{aligned} \dot{P}_o &= A_o P_o + P_o A_o^T + G_o Q_o Q_o^T - P_o H_O^T R_o^{-1} H_o P_o \\ \dot{\hat{x}} &= A_o\hat{x} + B_o u_o + P_o H_o^T R_o^{-1}(z_o - H_o\hat{x}) \ , \end{aligned} \tag{17}$$

where $P_o \in \mathbb{R}^{n\times n}$ is the gain matrix for the proposed Kalman-filter. Moreover, $Q_o \in \mathbb{R}^{n\times n}$ and $R_o \in \mathbb{R}^{p\times p}$ are the positive definite matrices for tuning the performance of the proposed Kalman-filter, based on the covariance of measurement noises.

Table 1. Variables in the Kalman-filter for linear motion of the mobile robots

Variable	WMR	Quadrotor
$\hat{x}$	$\begin{bmatrix}\hat{x}_w & \hat{y}_w & \hat{v}_w\end{bmatrix}^T$	$\begin{bmatrix}\vec{\hat{p}}_q & \vec{\hat{v}}_q\end{bmatrix}^T$
u_o	a_w^m	$\vec{a}_q^m$
z_o	$\begin{bmatrix}x_w^m & y_w^m\end{bmatrix}^T$	$\vec{p}_q^m$
A_o	$\begin{bmatrix}0 & 0 & \cos(\theta_w)\\ 0 & 0 & \sin(\theta_w)\\ 0 & 0 & 0\end{bmatrix}$	$\begin{bmatrix}\mathbf{0}_3 & \mathbf{I}_3\\ \mathbf{0}_3 & \mathbf{0}_3\end{bmatrix}$
B_o	$\begin{bmatrix}0 & 0 & 1\end{bmatrix}^T$	$\begin{bmatrix}\mathbf{0}_3\\ R_q\end{bmatrix}^T$
G_o	$\mathbf{I}_3$	$\mathbf{I}_6$
H_o	$\begin{bmatrix}1 & 0 & 0\\ 0 & 1 & 0\end{bmatrix}$	$\begin{bmatrix}\mathbf{I}_3 & \mathbf{0}_3\end{bmatrix}$

3.4.2. Angular motion

For the second part of the system's states which includes the angular displacement and the angular velocity of the mobile robot, a complementary filter proposed in Madgwick et al. (2011) can be used for eliminating the noise from the gyro measurement and also for computing the $3D$ angular displacements represented by Euler angles. The filtered noise-free data for angular motion is available by utilizing the commercial NGIMU sensor, which has the proposed complementary filter embedded in a compact module alongside with the low-cost IMU sensor. The derivation and implementation of the filter is presented in detail in Madgwick et al. (2011). Hence, here it is assumed that the measured data for angular displacements and the angular velocities are noise-free for the WMR and the quadrotor. The detailed information about the complementary filter can be referred to Madgwick et al. (2011).

4. Test Results

In this section, the data logged using MATLAB during the HIL tests are presented for both cases of testing including the WMR and the quadrotor. In both of the tests, the measurement noise on the measured states (i.e. z_o in Table 1) is a uniformly distributed noise with minimum value of $-\beta_0$ and the maximum value of β_0, where $\beta_0 \in \mathbb{R}$ is a constant value for the bounds of the noise. On the other hand, the type of measurement noise on the measured dynamics of the mobile robots (i.e. u_o in Table 1) is same as the previous case, but with the lower band of $-0.1 \times \beta_0$ and the maximum band of $0.1 \times \beta_0$. In fact, it is assumed that the noise on the acceleration measurement is 10 times lower than the noise on the position and speed measurements. This assumption is not far from reality, according to the available data sheets of the corresponding commercial

sensors. In addition, the constant parameters in the OAMFC algorithm are tuned with least possible efforts as shown in Table 2. In this table, $\mathbf{1}_h$ is a vector in $\mathbb{R}^{h\times 1}$ with all elements equal to one, where $h > 0$ is a constant scalar number. Moreover, $\mathbf{I}_h$ is an identity matrix in $\mathbb{R}^{h\times 1}$.

Table 2. Properties of the OAMFC algorithm with the Kalman-filter for two HIL tests. ($\gamma_0 = 10^{-6} \times \mathbf{1}_8$).

HIL test-1:WMR (each of linear and angular motions)	**HIL test-2:quadrotor**
$\Gamma_1 = diag([0.001; 10])$	$\Gamma_1 = diag([\gamma_0; 0.1; (10 \times \mathbf{1}_3)])$
$\Gamma_2 = diag([0.001; 0.1])$	$\Gamma_2 = diag([\gamma_0; (0.01 \times \mathbf{1}_4)])$
$\rho_1 = 0.001$	$\rho_1 = 0.001$
$\rho_2 = 0.001$	$\rho_2 = 0.001$
$Q = 0.01 \times \mathbf{I}_2$	$Q = 0.01 \times \mathbf{I}_{12}$
$R = \mathbf{I}_2$	$R = \mathbf{I}_{12}$
$k_1 = 1\ ,\ k_2 = 1$	$k_1 = 1\ ,\ k_2 = 1$
$Q_o = \mathbf{I}_2$	$Q_o = 10 \times \mathbf{I}_2$
$R_o = \mathbf{I}_2$	$R_o = 0.1 \times \mathbf{I}_2$

4.1. Case-1: WMR

The simulation results for the WMR are presented in Fig. (3) to Fig. (7). Here, the value of α_w is equal to 0.1. The desired trajectory in x-y plane is a rectangle with length of $5m$ and width of $3m$, as follows

$$\begin{aligned} x_d^w &= \begin{Bmatrix} 0 & , & t \le 10 \\ 5\ , & 10 < t \le 70 \\ 0 & , & 70 < t \end{Bmatrix}\ , \\ y_d^w &= \begin{Bmatrix} 0 & , & t \le 40 \\ 3\ , & 40 < t \le 100 \\ 0 & , & 100 < t \end{Bmatrix}\ . \end{aligned} \tag{18}$$

As it can be seen in the results, the errors for linear and rotational motions are bounded while converging to zero. Moreover, the adapted $\hat{A}$ and $\hat{g}$ and consequently the control signals are all bounded.

4.2. Case-2: Quadrotor

The simulation results for the quadrotor are depicted in Fig. (8) to Fig. (12). Here, the value of α_q is equal to 0.2. The desired trajectory is a helix along z-axis as follows

(all of the values are in m)

$$
\begin{aligned}
x_d^Q &= \begin{Bmatrix} 0 & , & t < 20 \\ 2\cos(0.2[t-20]) - 2 & , & t \geq 20 \end{Bmatrix} , \\
y_d^Q &= \begin{Bmatrix} 0 & , & t < 20 \\ 2\sin(0.2[t-20]) & , & t \geq 20 \end{Bmatrix} , \\
z_d^Q &= 0.2t \ .
\end{aligned} \tag{19}
$$

Considering the unknown disturbances and the measurement noise introduced in the previous section, the position errors are converged to zero, while the control signals are bounded.

4.3. Analysis on amplitudes of the external disturbance and the measurement noise

As it is shown in the test results (specifically Fig. 3 and Fig. 8), the proposed OAMFC algorithm incorporated with a Kalman-filter can provide an appropriate performance for the dynamic systems operating under the unknown external disturbances as well as bounded measurement noise. In this section, the effects of different values for amplitudes of the disturbances and the measurement noise are evaluated. Hence, the values for tuning parameters of the controller and the observer are fixed at the values presented in Table 2. The analysis includes two parts. First, the amplitudes of the external disturbances are supposed to be fixed and the effects of change in the amplitudes of the measurement noise are studied. At the second step, different values for amplitudes of the disturbances are considered, while the maximum values for random measurement noise are fixed. For the purpose of the evaluation, the sum of squared errors (SSE) for position of the WMR and the quadrotor is considered as follows

$$
\begin{aligned}
S_e^w &= \sum_{i=1}^{N_w} (\{x_w - x_d^w\}_i^2 + \{y_w - y_d^w\}_i^2) \ , \\
S_e^Q &= \sum_{i=1}^{N_q} (\{x_q - x_d^Q\}_i^2 + \{y_q - y_d^Q\}_i^2 + \{z_q - z_d^Q\}_i^2) \ ,
\end{aligned} \tag{20}
$$

where N_w and N_q are the total number of time steps in HIL tests for WMR and quadrotor, respectively. The analysis on amplitude of the measurement noise is proposed in Table 3 for both HIL tests, according to the values of SSE. These values are derived for the cases where the amplitudes of the disturbance are fixed at $\alpha_w = 0.1$ and $\alpha_q = 0.2$. In addition, the values of SSE for the case where the value of $\beta_0 = 0.5$ is constant, is presented in Table 4. These values can be used for evaluating the effects of the disturbance amplitude on the performance of the proposed OAMFC algorithm. Since the test results are affected by the random nature of measurement noise, the presented data in Table 3 and Table 4 are the average values for 10 sets of consecutive test results. According to the data provided in these tables, we can see that the performance of the OAMFC algorithm degrades with increasing the amplitude of the disturbance. The performance can be improved by tuning the values in Γ_1 and Γ_2 matrices. On the other hand, the effect of increasing the amplitude of the measurement noise is minor on the performance of the OAMFC algorithm. It is also an indication

for the usefulness of the Kalman-filter.

Table 3. Values of SSE for different amplitudes of the measurement noise ($\alpha_w = 0.1$, $\alpha_q = 0.2$).

Value of SSE	$\beta_0 = 0.1$	$\beta_0 = 0.5$	$\beta_0 = 0.9$
S_e^w	$2.59e4$	$2.81e4$	$2.93e4$
S_e^Q	$4.66e3$	$4.68e3$	$4.70e3$

Table 4. Values of SSE for different amplitudes of the external disturbance ($\beta_0 = 0.5$).

Value of SSE	$\alpha_w = 0.05$	$\alpha_w = 0.1$	$\alpha_w = 0.2$
S_e^w	$2.48e4$	$2.81e4$	$6.05e4$
	$\alpha_q = 0.1$	$\alpha_q = 0.2$	$\alpha_q = 0.4$
S_e^Q	$3.96e3$	$4.66e3$	$7.32e3$

5. Conclusion

In this paper, the recently proposed OAMFC algorithm is tested in a HIL test platform to evaluate the performance of the algorithm one step before the prototyping on real mobile robots. Detailed simulation results with a complete comparison study is presented in our previous works. Here, we investigated the challenges encountered during the real implementation of the algorithm. The algorithm is embedded in a development board, while a dynamic model for the mobile robot is implemented in another development board. The boards are communicating emulated measured data and the corresponding generated control signals through I^2C communication protocol. The implementation of the algorithm is optimized in a way that the least value for sampling time can be achieved. MATLAB is utilized for logging the data during the HIL test. Provided results show that the performance of the OAMFC algorithm is satisfactory under existence of the bounded unknown external disturbances and also bounded measurement noise. The OAMFC algorithm is validated for implementing on the real mobile robots with much more confirmation. It should be noted that *a priori* information about the dynamic characteristic of the mobile robot (like mass, inertia and dimensions) are not required for implementing the proposed OAMFC algorithm. Such convenience illustrates its salient feature in minimizing control synthesis proving it attractive for wide practical.

Acknowledgment

This work is supported by a Research University (RUi) 1001/PELECT/8014029 awarded from Universiti Sains Malaysia. Besides, the PhD studies of the first author is under a TWAS-USM Postgraduate Fellowship.

References

Fliess M. and C. Join (2013). Model-free control. *International Journal of Control*, *86*(12): 2228-2252.

R.-E. Percup, M. B. Radac, R.-C. Roman and E. M. Petriu, "Model-free sliding mode control of nonlinear systems: algorithms and experiments," *Information Sciences*, Vol. 381, pp. 176-192, 2017.

K. G. Vamvoudakis, D. Vrabie and F. L. Lewis, "Online adaptive algorithm for optimal control with integral reinforcement learning," *International Journal of Robust and Nonlinear Control*, Vol. 24, Issue 17, pp. 2686-2710, 2014.

Y. Zhu, D. Zhao and X. Li, "Using reinforcement learning techniques to solve continuous-time non-linear optimal tracking problem without system dynamics," *IET Control Theory and Applications*, Vol. 10, Issue 12, pp. 1339-1347, 2016.

A. Safaei and M. N. Mahyuddin, "Adaptive model-free control based on an ultra-local model with model-free parameter estimations for a generic SISO system", *IEEE Access*, Vol. 6, pp. 4266-4275, 2018.

A. Safaei and M. N. Mahyuddin, "Optimal model-free control for a generic MIMO nonlinear system with application to autonomous mobile robots", *International Journal of Addaptive Control and Signal Processing*, Vol. 32, Issue 6, pp. 792-815, 2018.

Safaei A., Y. C. Koo and M. N. Mahyuddin (2017). Adaptive model-free control for robotic manipulators. in: *Proc. IEEE International Symposium on Robotics and Intelligent Sensors (IRIS2017)*, Ottawa, Canada, pp. 7-12.

A. Safaei and M. N. Mahyuddin, "An optimal adaptive model-free control with a Kalman-filter-based observer for a generic nonlinear MIMO system", in *Proc. 2017 IEEE 2nd International Conference on Automatic Control and Intelligent Systems (I2CACIS2017)*, Kota Kinabalu, Malaysia, October 2017, pp. 56-61.

A. Safaei and M. N. Mahyuddin, "Application of the Optimal Adaptive Model-Free Control Algorithm on an Autonomous Underwater Vehicle", in *Proc. IEEE International Conference on Advanced Robotics and Mechatronis (ICARM2018)*, Singapore, July 2018.

R. Verma, D. D. Vecchio and H. K. Fathy, "Development of a Scaled Vehicle With Longitudinal Dynamics of an HMMWV for an ITS Testbed," in *IEEE/ASME Transactions on Mechatronics*, vol. 13, no. 1, pp. 46-57, 2008.

G. D. White, R. M. Bhatt, C. P. Tang and V. N. Krovi, "Experimental Evaluation of Dynamic Redundancy Resolution in a Nonholonomic Wheeled Mobile Manipulator," in *IEEE/ASME Transactions on Mechatronics*, vol. 14, no. 3, pp. 349-357, 2009.

M. M. Michałek and M. Kiełczewski, "The Concept of Passive Control Assistance for Docking Maneuvers With N-Trailer Vehicles," in *IEEE/ASME Transactions on Mechatronics*, vol. 20, no. 5, pp. 2075-2084, 2015.

Y. Z. Li, S. J. Zhu, Y. Li and Q. Lu, "Temperature Prediction and Thermal Boundary Simulation Using Hardware-in-Loop Method for Permanent Magnet Synchronous Motors," in *IEEE/ASME Transactions on Mechatronics*, vol. 21, no. 1, pp. 276-287, 2016.

C. Qi, A. Ren, F. Gao, X. Zhao, Q. Wang and Q. Sun, "Compensation of Velocity Divergence Caused by Dynamic Response for Hardware-in-the-Loop Docking Simulator," in *IEEE/ASME Transactions on Mechatronics*, vol. 22, no. 1, pp. 422-432, 2017.

C. Chéron, A. Dennis, V. Semerjyan and Y. Chen, "A multifunctional HIL testbed for multi-rotor VTOL UAV actuator," in *Proceedings of 2010 IEEE/ASME International Conference on Mechatronic and Embedded Systems and Applications*, Qingdao, ShanDong, China, 2010, pp. 44-48.

M. K. Bayrakceken, M. K. Yalcin, A. Arisoy and A. Karamancioglu, "HIL simulation setup for attitude control of a quadrotor," in *Proceddings of 2011 IEEE International Conference on Mechatronics*, Istanbul, Turkey, 2011, pp. 354-357.

H. S. Khan and M. B. Kadri, "Position control of quadrotor by embedded PID control with hardware in loop simulation," in *Proceedings of 17th IEEE International Multi Topic Conference*, Karachi, Pakistan, 2014, pp. 395-400.

H. K. Khalil, *Nonlinear Systems*, Third Edition, Prentice Hall Inc., 2002.

Jin. Q. Cui, S. Lai, X. Dong and B. M. Chen, "Autonomous navigation of UAV in foliage environment," *Journal of Intelligent and Robotic Systems: Theory and Applications*, Vol. 84, Issue 1-4, pp. 259-276, 2016.

Frank. L. Lewis, L. Xie and D. Popa, *Optimal and robust estimations*, second edition, CRC Press, Taylor and Francis Group, 2007.

Sebastian O.H. Madgwick, A.J.L Harrison and R. Vaidyanathan, "Estimation of IMU and MARG orientation using a gradient descent algorithm," in proceddings of 2011 IEEE International Conference on Rehabilitation Robotics, Switzerland, June 2011, pp. 1-7.

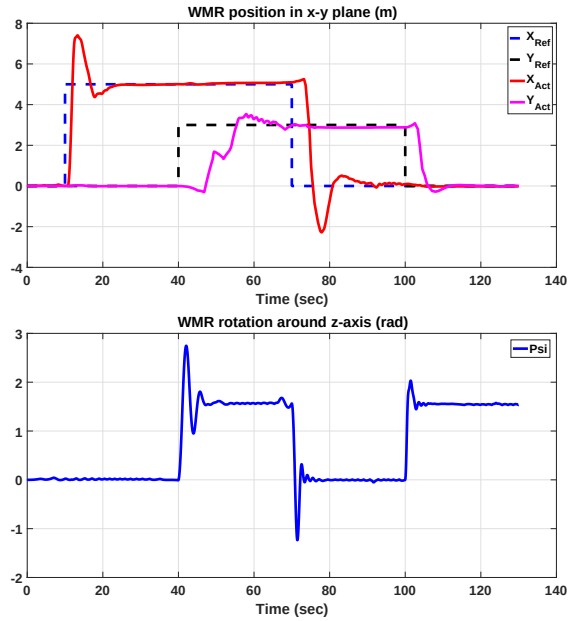

Figure 3. Case-1: The x-y position and z-axis rotation of the WMR.

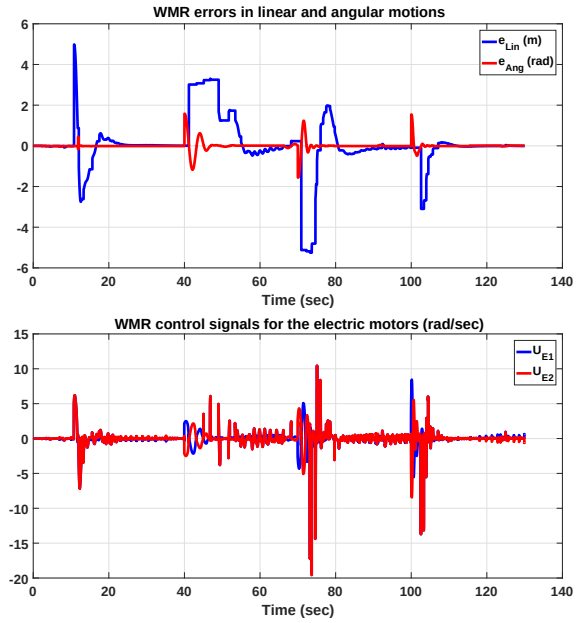

Figure 4. Case-1: The errors in linear and angular motions of the WMR (top); and the control signals for the electric motors of the WMR (bottom).

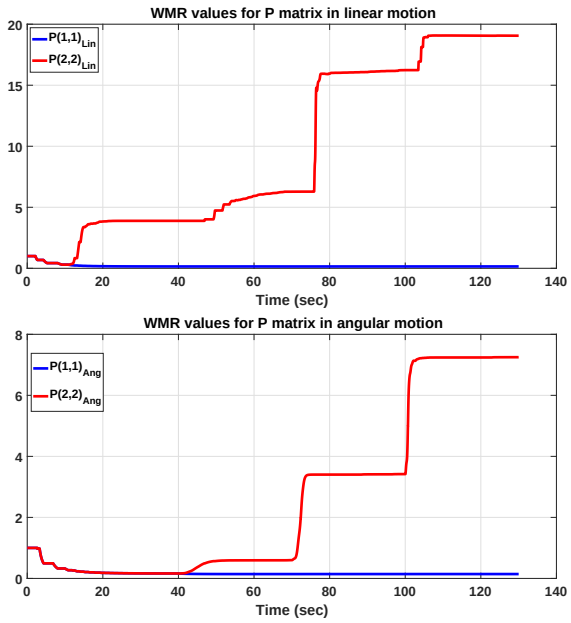

Figure 5. Case-1: The values of P matrix for the WMR in linear and angular motions.

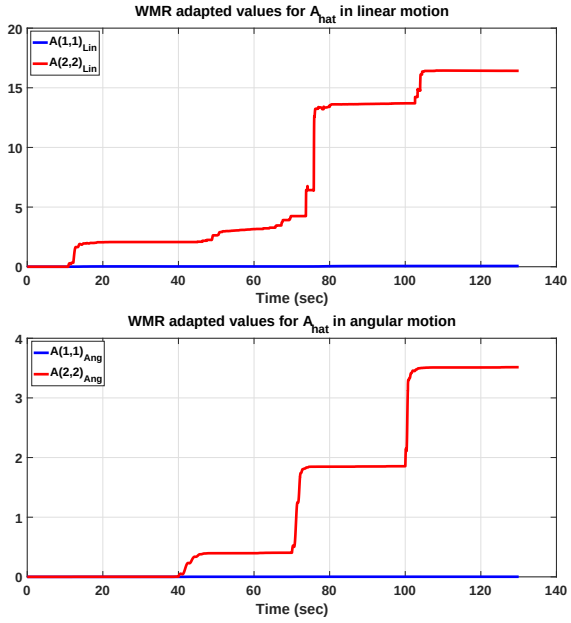

Figure 6. Case-1: The adapted values of unknown $\hat{A}$ for the WMR in linear and angular motions.

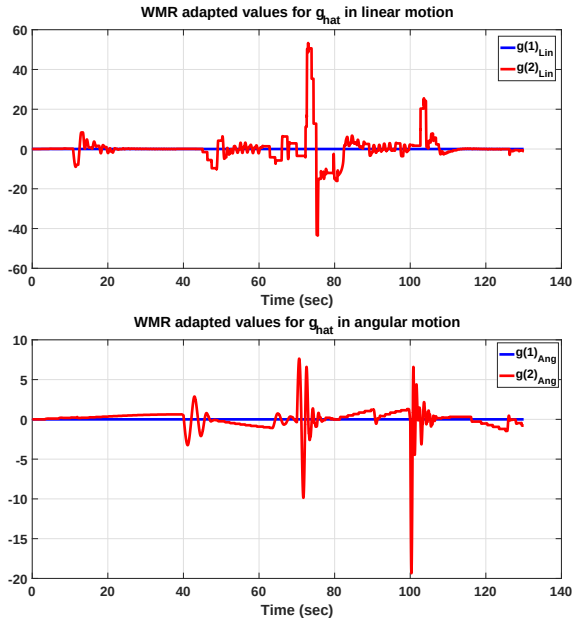

Figure 7. Case-1: The adapted values of unknown $\hat{g}$ for the WMR in linear and angular motions.

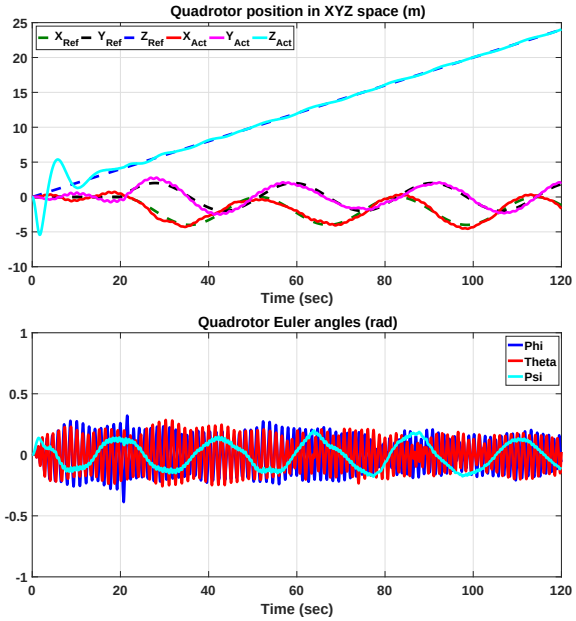

Figure 8. Case-2: The quadrotor position in 3D space and its Euler angles.

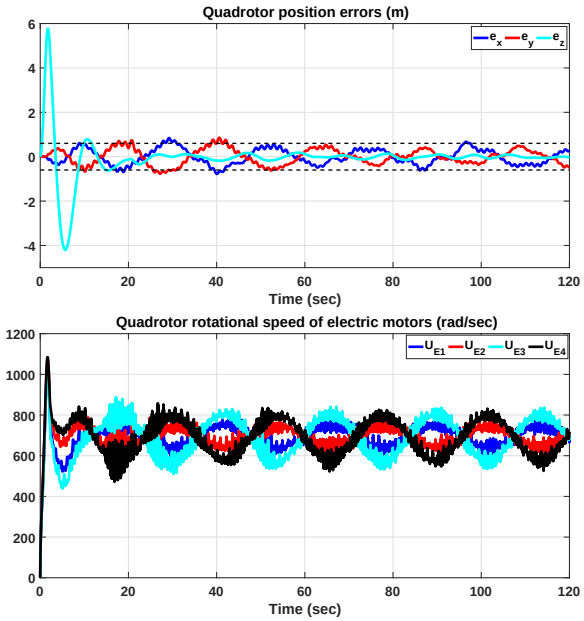

Figure 9. Case-2: The errors in position of the quadrotor (top); and the control signals for the electric motors of the quadrotor (bottom). The values of errors are bounded with absolute upper-bound of 0.6 .

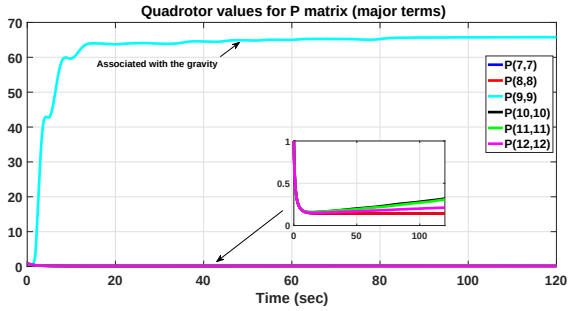

Figure 10. Case-2: The values of P matrix for quadrotor.

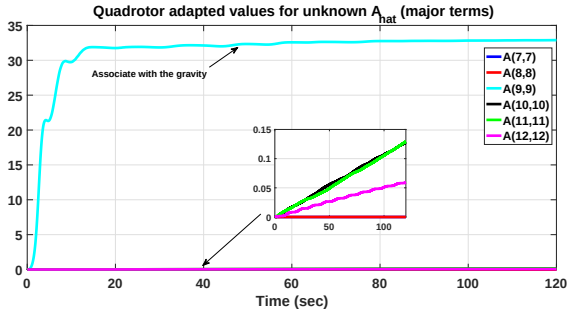

Figure 11. Case-2: The adapted values of unknown $\hat{A}$ for the quadrotor.

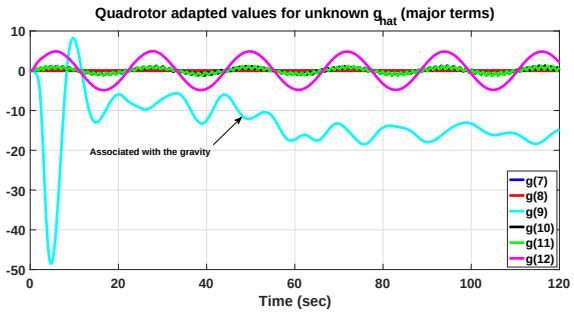

Figure 12. Case-2: The adapted values of unknown $\hat{g}$ for the quadrotor.