Adaptive Intelligent Controller Design for a New Quadrotor Manipulation System

Ahmed Khalifa[1], Mohamed Fanni[2], Ahmed Ramadan[3] and Ahmed Abo-Ismail[4]
[1,4]Department of Mechatronics and Robotics Engineering
Egypt-Japan University of Science and Technology, New Borg-El-Arab city, Alexandria, Egypt
[2]Department of Production Engineering and Mechanical Design
Mansoura University, Mansoura, Egypt
[3]Department of Computer and Automatic Control
Tanta University, Tanta, Egypt
Email: ahmed.khalifa@ejust.edu.eg , mfanni@mans.edu.eg , ahmed.ramadan@ejust.edu.eg and aboismail@ejust.edu.eg

***Abstract*—This paper presents the design of two controllers namely, Direct Fuzzy Logic controller and Fuzzy Model Reference Learning Control applied to a new aerial manipulation system called "Quadrotor-Manipulator System". This system consists of 2-link manipulator connected to the bottom of a quadrotor. The dynamic model of this system is derived taking into account the effect of adding a payload to the manipulator. The performance of the proposed controllers are compared with that of the previously developed controller based on Feedback Linearization technique. All controllers are tested regarding their ability to stabilize the system and track desired trajectories under the effects of picking and placing a payload as well as changing the system operating region. Finally, the system equations of motion and the control laws are simulated using MATLAB/SIMULINK. The simulation results indicate the outstanding performance of the Fuzzy Model Reference Learning Control.**

Keywords*—Aerial manipulation; Quadrotor-Manipulator; 2-link manipulator; Feedback linearization; Demining devices; Fuzzy logic control; Fuzzy model reference learning control

I. INTRODUCTION

In recent years, extensive research works on Unmanned Aerial Vehicles (UAVs) have been done in [1]–[6]. UAVs offer possibilities of speed and access to regions that are otherwise inaccessible to ground robotic vehicles. Quadrotor vehicles possess certain essential characteristics, which highlight their potential for use in search and rescue applications.

An aerial manipulator was presented in [1]. This system consists of a gripper that connected to the bottom of a quadrotor. This design of the aerial manipulator enables the quadrotor to interact with the environment. So, one can get an entire new set of applications. First, allowing robots to fly and perch on rods or beams increases the endurance of their missions (e.g. for its battery charging). Second, the ability to grasp and manipulate objects allows robots to access payloads that cannot be manipulated by ground robots. With this quadrotor system, a 4 DOF manipulator was established.

The authors introduced a new quadrotor manipulation system in [7] that consists of two-link manipulator with two revolute joints attached to the bottom of a quadrotor. The two axes of the revolute joints of the manipulator are perpendicular to each other. With this new system, the capability of manipulating objects with arbitrary location and orientation is achieved because the degrees of freedom of the end effector (in our case, it is a gripper) are increased from 4 to 6. On the other hand, the manipulator provides sufficient distance between the quadrotor and the object location. This system has a lot of potential applications such as demining applications (e.g Improvised Explosive Device Disposal), performing maintenance for a bridge or building, hazardous material handling and removal. In this work [7], the modelling of the system is proposed but without taking into consideration the effect of carrying a payload by the gripper. Also, a controller design based on the feedback linearization technique was presented and tested to achieve stabilization and trajectory tracking.

In this paper, the dynamic model of the quadrotor manipulation system is modified to study the effect of picking and placing the payload. Then, two control techniques are designed and tested in order to stabilize the system and track a desired trajectory under the effect of adding and releasing a payload as well as under the effect of changing the system operating region. These controllers are Direct Fuzzy Logic Control (DFLC) and adaptive fuzzy control based on Fuzzy Model Reference Learning Control (FMRLC). Also their performance are compared to that based on Feedback Linearization (FBL) control technique.

This paper is organized as follows. The modeling of the system is described in section II. Section III introduces the control strategy of the system, while DFLC and FMRLC are presented in sections IV and V respectively. The simulation results using MATLAB/SIMULINK are presented in section VI. Finally, the main contributions are concluded in section VII.

II. MODIFIED MODELING OF THE SYSTEM

3D CAD model of the proposed system is shown in Fig. 1. The system consists of two-link manipulator attached to the bottom of a quadrotor. The manipulator has two revolute joints. The axis of joint 1 (z_0 in Fig. 2) is parallel to one in-plane axis of the quadrotor (x in Fig. 2) and perpendicular to the axis of joint 2. Also, the axis of joint 2 (z_1 in Fig. 2) is parallel to the other in-plane axis of the quadrotor (y in Fig. 2) at home (extended) configuration. Therefore the end effector can perform any arbitrary position and orientation. So, a 6 DOF aerial manipulator is obtained.

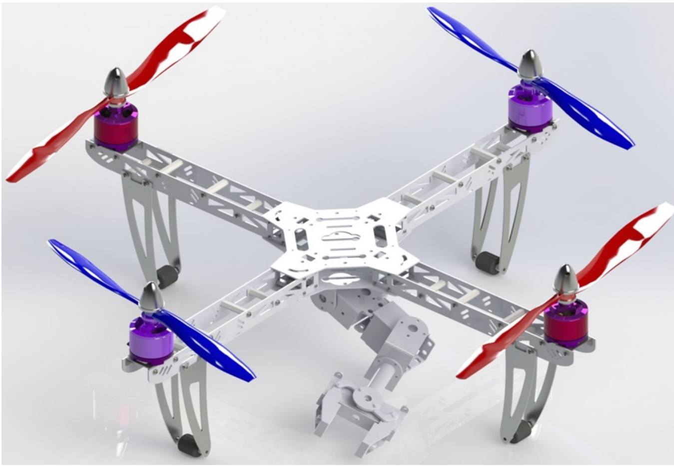


Fig. 1. 3D CAD Model of the New Quadrotor Manipulation System

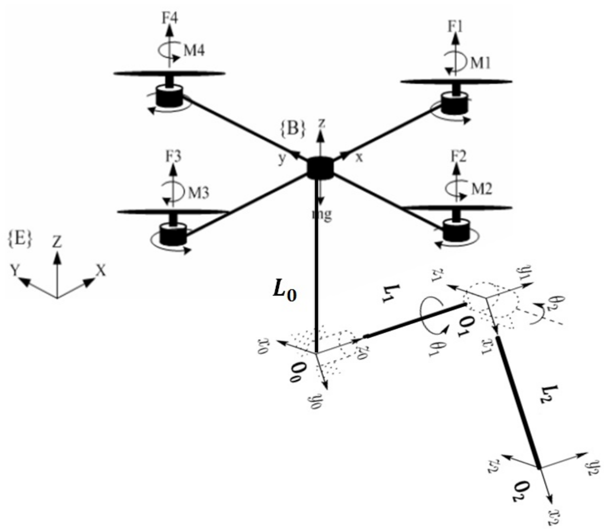


Fig. 2. Schematic of Quadrotor Manipulation System Frames

Derivation of the equations of motion for the quadrotor manipulation system is presented in details in [7].

A. System Kinematics

Fig. 2 presents a sketch of the quadrotor-manipulator system with the relevant frames. The orientation of the quadrotor is represented through Euler Angles. A rigid body is completely described by its position and orientation with respect to a reference frame {E}, O_I-X Y Z, that is earth-fixed. Let

$$R_I^B = \begin{bmatrix} C(\psi)C(\theta) & S(\psi)C(\theta) & -S(\theta) \\ -S(\psi)C(\phi)+S(\psi)S(\theta)C(\psi) & C(\psi)C(\phi)+S(\psi)S(\theta)S(\phi) & C(\theta)S(\phi) \\ S(\psi)S(\phi)+C(\psi)S(\theta)C(\phi) & -C(\psi)S(\phi)+S(\psi)S(\theta)C(\phi) & C(\theta)C(\phi) \end{bmatrix} \quad (1)$$

be the rotation matrix expressing the transformation from the inertial frame to the body-fixed frame{B}, O_B-x y z, where ϕ, θ, and ψ are Euler angles. Note that $C(.)$ and $S(.)$ in (1) are short notations for $cos(.)$ and $sin(.)$. Let us assume a small angles of ϕ and θ, then the corresponding time derivatives of Euler angles are equal to the body-fixed angular velocity components.

In Fig. 2 the frames satisfy the Denavit-Hartenberg convention [8]. The position and orientation of the end effector relative to the body-fixed frame is easily obtained by multiplying appropriate homogeneous transformation matrices [7], [8].

B. System Dynamics

Applying Newton Euler algorithm [9] to the manipulator considering that the link (with length L_0) that is fixed to the quadrotor is the base link, one can get the equations of motion of the manipulator as well as the interaction forces and moments between the manipulator and the quadrotor. The effect of adding a payload to the manipulator will appear in the parameters of its end link, link 2, (e.g. mass, center of gravity, and inertia matrix). Therefore, the payload will change the overall system dynamics.

The equations of motion of the manipulator are:

$$M_1\ddot{\theta}_1 = T_{m_1} + N_1 \quad (2)$$

$$M_2\ddot{\theta}_2 = T_{m_2} + N_2 \quad (3)$$

where, T_{m_1} and T_{m_2} are the manipulator actuators' torques. M_1, M_2, N_1, and N_2 are nonlinear terms and they are functions in the system states as described in [7].

The Newton Euler method are used to find the equations of motion of the quadrotor after adding the forces/moments applied by the manipulator are:

$$m\ddot{X} = T(C(\psi)S(\theta)C(\phi) + S(\psi)S(\phi)) + F^I_{m,q_x} \quad (4)$$

$$m\ddot{Y} = T(S(\psi)S(\theta)C(\phi) - C(\psi)S(\phi)) + F^I_{m,q_y} \quad (5)$$

$$m\ddot{Z} = -mg + TC(\theta)C(\phi) + F^I_{m,q_z} \quad (6)$$

$$I_x\ddot{\phi} = \dot{\theta}\dot{\phi}(I_y - I_z) - I_r\dot{\theta}\overline{\Omega} + T_{a_1} + M^B_{m,q_\phi} \quad (7)$$

$$I_y\ddot{\theta} = \dot{\psi}\dot{\phi}(I_z - I_x) + I_r\dot{\phi}\overline{\Omega} + T_{a_2} + M^B_{m,q_\theta} \quad (8)$$

$$I_z\ddot{\psi} = \dot{\theta}\dot{\phi}(I_x - I_y) + T_{a_3} + M^B_{m,q_\psi} \quad (9)$$

where F^I_{m,q_x}, F^I_{m,q_y}, and F^I_{m,q_z} are the interaction forces from the manipulator to the quadrotor in X,Y, and Z directions defined in the inertial frame and M^B_{m,q_ϕ} , M^B_{m,q_θ} , and M^B_{m,q_ψ} are the interaction moments from the manipulator to the quadrotor around X, Y, and Z directions defined in the inertial frame.

The variables in (4-9) are defined as follows: T is the total thrust applied to the quadrotor from all four rotors, and is given by:

$$T = \sum_{i=1}^{4}(F_j) = \sum_{i=1}^{4}(b\Omega_j^2) \quad (10)$$

where F_j is the thrust force from rotor j, $\Omega_j$is the angular velocity of rotor j and b is the thrust coefficient. T_{a_1}, T_{a_2}, and T_{a_3} are the three input moments about the three body axes, and are given as:

$$T_{a_1} = d(F_4 - F_2) \quad (11)$$

$$T_{a_2} = d(F_3 - F_1) \quad (12)$$

$$T_{a_3} = K_d(-\Omega_1^2 + \Omega_2^2 - \Omega_3^2 + \Omega_4^2) \quad (13)$$

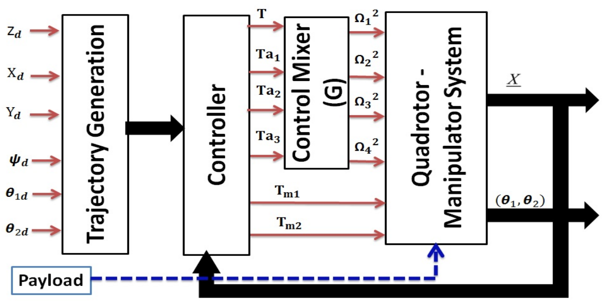


Fig. 3. Block Diagram of the Control System

d is the distance between the quadrotor center of mass and the rotation axis of the propeller and K_d is the drag coefficient.

$$\overline{\Omega} = \Omega_1 - \Omega_2 + \Omega_3 - \Omega_4 \tag{14}$$

I_r is the rotor inertia. I_f is the inertia matrix of the vehicle around its body-frame assuming that the vehicle is symmetric about x-, y- and z-axis.

From the equations of the translation dynamics (4-6), one can extract the expressions of the high-order nonholonmic constraints:

$$\sin(\phi) = \frac{(\ddot{X} - F^I_{m,q_x})\sin(\psi) - (\ddot{Y} - F^I_{m,q_y})\cos(\psi)}{\sqrt{(\ddot{X} - F^I_{m,q_x})^2 + (\ddot{Y} - F^I_{m,q_y})^2 + (\ddot{Z} + g - F^I_{m,q_z})^2}} \tag{15}$$

$$\tan(\theta) = \frac{(\ddot{X} - F^I_{m,q_x})\cos(\psi) + (\ddot{Y} - F^I_{m,q_y})\sin(\psi)}{\ddot{Z} + g - F^I_{m,q_z}} \tag{16}$$

where F^I_{m,q_x}, F^I_{m,q_y}, and F^I_{m,q_z} are functions of the system states and there derivatives.

III. System Controller Design

Quadrotor is an under-actuated system, because it has four inputs (angular velocities of its four rotors) and six variables to be controlled. By observing the operation of the quadrotor, one can find that the movement in x- direction is based on the pitch rotation, θ. Also the movement in y- direction is based on the roll rotation, ϕ. Therefore; motion along X- and Y-axes will be controlled through controlling θ and ϕ.

Fig. 3 presents a block diagram of the proposed control system. The control design criteria are achieving system stability and a zero trajectory tracking error under the effect of:

- Picking and placing a payload.
- Changing the operating region of the system.

The matrix G of the control mixer is used to transform the assigned thrust force and moments of the quadrotor (the control signals) from the controller block into assigned angular velocities of the four rotors. This matrix can be derived from (10-13) as in [7].

Feedback linearization based PID controller is designed and tested regarding tracking desired trajectories of the quadrotor-manipulator system in [7]. Fig. 4 presents the block diagram of this control techniques.

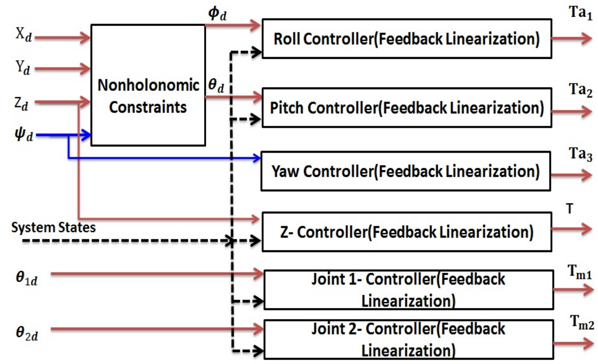


Fig. 4. Details of the Controller Block in Case of Feedback Linearization

In Fig. 4, the nonholonmic constraints are used to determine the desired trajectories of θ and ϕ from the desired trajectories of X, Y, Z, ψ, θ_1, and θ_2 and their derivatives. Then feedback linearization controllers are used to obtain a zero tracking errors for θ, ϕ, Z, ψ, θ_1 and θ_2. Putting subscript d to all variables in (15 and 16), then ϕ_d and θ_d can be obtained. The details of this controller design are presented in [7].

IV. Direct Fuzzy Logic Control

Recently, fuzzy logic control [10], [11] has become an alternative to conventional control algorithms to deal with complex processes and combine the advantages of classical controllers and human operator experience.

An intelligent controller for a quadrotor was designed and presented in [12]. In this work, a modification of this technique is done and used to control the quadrotor-manipulator system to achieve the required objectives mentioned in Section III.

In Fig. 5, three fuzzy controllers are designed to control the quadrotor's roll (ϕ), pitch (θ) and yaw(ψ) angles, denoted by FLC_ϕ, FLC_θ, and FLC_ψ, respectively, with the former two serving as attitude stabilizers. Three fuzzy controllers, FLC_x, FLC_y and FLC_z, are further designed to control the quadrotor's position. Also two fuzzy controllers FLC_{θ_1} and FLC_{θ_2} are designed to control the two joints' angles of the manipulator.

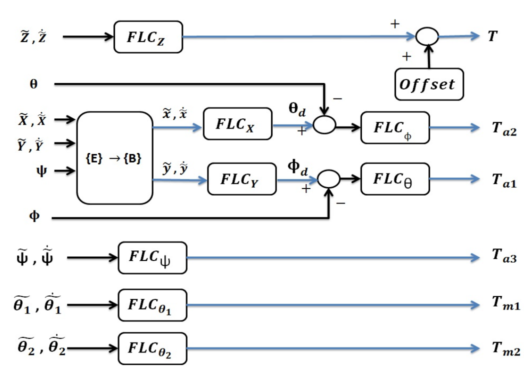


Fig. 5. Details of the Controller Block in Case of DFLC

All eight fuzzy controllers have similar inputs that are:

- The error $e = (\tilde{.}) = (.)_d - (.)$, which is the difference between the desired signal $(.)_d$ and its actual value (.). This input is normalized to the interval [-1, +1].
- The error rate c, which is normalized to the interval [-3, +3].

In this control strategy, the desired pitch and roll angles, θ_d and ϕ_d , are not explicitly provided to the controller. Instead, they are continuously calculated by controllers FLC_x and FLC_y in such a way that they stabilize the quadrotor's attitude. First, we convert the error and its rate of X and Y that is defined in the inertial frame into their corresponding values defined in the body frame. This conversion is done using the transformation matrix defined in (1) assuming small angles (ϕ and θ) approximation as following:

$$\dot{\tilde{x}} = \dot{\tilde{X}} \cos(\psi) + \dot{\tilde{Y}} \sin(\psi) \tag{17}$$

$$\dot{\tilde{y}} = \dot{\tilde{X}} \sin(\psi) - \dot{\tilde{Y}} \cos(\psi) \tag{18}$$

The input and output membership functions of each FLC are tuned and chosen to be three symmetric triangular shaped functions with the linguistic values N (Negative), Z (Zero), and P (Positive). Also the input and output scaling factors for the error, change of error, and fuzzy output (K_{e_i}, K_{c_i}, and K_{u_i}; i = x, y, z, ϕ, θ, ψ, θ_1, θ_2) of each FLC are tuned such that required performance is obtained.

The rule base of each FLC block is the same and is designed to provide a PD-like fuzzy controller. A Mamdani fuzzy inference method is used with a min-max operator for the aggregation and the center of gravity method for defuzzification.

There is a need to add an Offset value to the control signal from the FLC_Z (T) in order to counter balance the weight of the quadrotor. This value is equal to the total weight of the quadrotor.

It is important to note that this control scheme does not depend on the kinematic and dynamic equations derived in Section II. The fuzzy controllers are designed in light of the knowledge acquired on the quadrotor's behavior and from its dynamic model.

V. Adaptive Fuzzy Logic Control

In this section, an adaptive fuzzy logic control based on "fuzzy model reference learning controller"is designed to control the quadrotor-manipulator system. This control technique is presented in details in [10], [13], [14].

The main drawback of fuzzy controllers is the large amount of parameters to be tuned. Also, the direct FLC designed in section IV needs to retune its parameters in each operation region. Moreover, the fuzzy controller constructed for the nominal plant may later perform inadequately if significant and unpredictable plant parameter variations, or environmental disturbances occur [13].

In this work, a learning control algorithm employs a reference model to provide closed-loop performance feedback for tuning a fuzzy controller's knowledge-base. These performance objectives are characterized via the reference model.

The control system design is the same as in Fig. 5 by replacing each of the FLC_z, FLC_ϕ, FLC_θ, FLC_ψ, FLC_{θ_1} and FLC_{θ_2} block with the block shown in Fig. 6. However, there is no need for the offset value that is used in Fig. 5 because the FMRLC can compensate the quadrotor weight. The blocks of FLC_x and FLC_y are still the same because there is no need for adaptation here, since these blocks are used to map the relation between the error in X and Y directions into the required roll and pitch motions based on the operation of the quadrotor.

In the following subsections, the individual blocks in Fig. 6 will be described briefly.

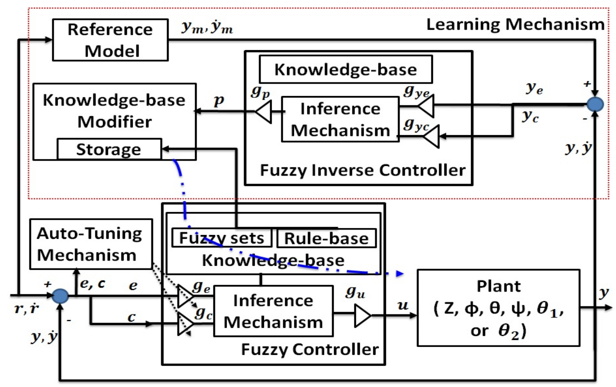


Fig. 6. Functional Block Diagram for the FMRLC

A. The Fuzzy Controller

The plant in Fig. 6 has output y (Z, ϕ, θ, ψ, θ_1, or θ_2) and an input u. Scaling gains g_e, g_c, and g_u for the error, e, change in error, c, and controller output, u, are used respectively, such that the universe of discourse of all inputs and outputs are the same and equal to [-1, 1].

The inputs membership functions are chosen to be 11 symmetric triangular-shaped functions. The initial rule base elements are set to zeros. The output membership functions are symmetric triangular-shaped functions and all centered at zero. They are what the FMRLC will automatically tune. Mamdani fuzzy inference method is used with a min-max operator for the aggregation. The standard center of gravity is used as a defuzzification technique.

B. The Reference Model

The reference model is used to quantify the desired performance. A 1^{st} order model is chosen as the reference model:

$$\frac{y_m(s)}{r(s)} = \frac{1}{\tau_{c_i} S + 1} \tag{19}$$

where τ_{c_i} (i = z, ϕ, θ, ψ, θ_1, and θ_2) is the time constant of the reference model.

C. The Learning Mechanism

The learning mechanism tunes the rule-base of the direct fuzzy controller so that the closed-loop system behaves like the reference model. The learning mechanism consists of two parts, fuzzy inverse model and knowledge-base modifier.

1) The Fuzzy Inverse Model: The fuzzy inverse model performs the function of mapping y_e (representing the deviation from the desired behavior), to changes in the process inputs p that are necessary to force y_e to zero. The fuzzy inverse model shown in Fig. 6 contains scaling gains, g_{y_e}, g_{y_c}, and g_p. Also, it has 11 symmetric triangular-shaped membership functions for the input and output universes of discourse. Mamdani fuzzy inference method is used with a min-max operator for the aggregation and the standard center of gravity is used as a defuzzification technique. The rule base of the fuzzy inverse model is shown in Table that is can be got from [13].

2) The Knowledge-Base Modifier: Given the information about the necessary changes in the input, which are represented by p, to force the error y_e to zero, the knowledge-base modifier changes the rule-base of the fuzzy controller so that the previously applied control action will be modified by the amount p.

D. Auto-Tuning Mechanism

In the standard FMRLC design, the system performance is degraded with variation in the desired input value. An auto-tuning mechanism is used in [14] to tune g_e and g_c gains online as following: Let the maximum of each fuzzy controller input (e, c) over a time interval of the last T_a seconds be denoted by $max_{T_a}\{e\}$ and $max_{T_a}\{c\}$. Then this maximum value is defined as the gain of each input e and c so that,

$$g_e = \frac{1}{max_{T_a}\{e\}}, \quad and \qquad g_c = \frac{1}{max_{T_a}\{c\}} \tag{20}$$

The learning mechanism must be operated at a higher rate than the auto-tuning mechanism.

VI. Simulation Results

The system equations of motions and the control laws of the three control techniques are simulated using MATLAB/SIMULINK program. Parameters of the system are listed in [7]. Quintic Polynomial Trajectories [8] are used as the reference trajectories for X, Y, Z, ψ, θ_1, and θ_2. Those types of trajectories have sinusoidal acceleration which is better in order to avoid vibration modes. All trajectories have the same characteristics, such that the initial position = 0, the final position = 1, except ψ trajectory with final position = 0, the initial and final velocities and accelerations equal to zero, and the final time is 10 s and the simulation time is 60 s. The controller parameters of the feedback linearization, DFLC, and FMRLC controllers are given in [7], Table I, and Table II respectively. Those parameters are tuned to get the required system performance.

The three controller are tested to stabilize and track the desired trajectories under the effect of picking a payload of value 100 g at instant 20 s and placing it at instant 40 s. The simulation results are presented in Fig. 7. The performance of the controllers in the directions of X, Y, and Z is the same and θ_1 and θ_2 is the same, so only the results of X, ψ, and θ_2 are presented. These results show that the controller design based on feedback linearization can track the desired trajectories before picking the payload but at the instant of picking and then holding the payload, it fails to track the desired trajectories and the system becomes unstable even if the payload is released. Both DFLC and FMRLC are able to track the desired trajectories before, during picking, holding, and placing the payload but the DFLC produces a steady state error during the period of holding the payload. DFLC suffers from the necessity of calibrating and determining the offset value which is affected by payload value and cannot be estimated accurately.

TABLE I. DFLC Parameters

Par.	Value	Par.	Value
$[K_{e_x} K_{c_x} K_{u_x}]$	$[.0035, .01, 35]$	$[K_{e_y} K_{c_y} K_{u_y}]$	$[.0035, .01, 35]$
$[K_{e_z} K_{c_z} K_{u_z}]$	$[.1, .5, 100]$	$[K_{e_\psi} K_{c_\psi} K_{u_\psi}]$	$[.05, .1, 4]$
$[K_{e_\phi} K_{c_\phi} K_{u_\phi}]$	$[.1, .1, 14]$	$[K_{e_{\theta_1}} K_{c_{\theta_1}} K_{u_{\theta_1}}]$	$[.05, .05, 9]$
$[K_{e_\theta} K_{c_\theta} K_{u_\theta}]$	$[.1, .1, 14]$	$[\ [K_{e_{\theta_2}} K_{c_{\theta_2}} K_{u_{\theta_2}}]$	$[.05, .05, 9]$

TABLE II. FMRLC Parameters

Par./Val.	Z	ϕ	θ	ψ	θ_1	θ_2
$g_{e-initial}$	1/5	2	2	1	1/6	1/6
$g_{c-initial}$	1/10	1	1	100	1	1
g_u	45.9	2.9	2.9	0.19	0.94	0.94
g_{y_e}	1/60	1/3	1/3	1	1/12.5	1/6.2
g_{y_c}	1/60	1/3	1/3	10	1/12.5	1/6.2
g_p	0.45	0.029	0.029	0.0038	0.0047	0.0188
$\tau_c(s)$	1.5	0.001	0.001	0.01	0.1	0.1
$T_a(s)$	0.1	0.05	0.05	0.05	0.05	0.05

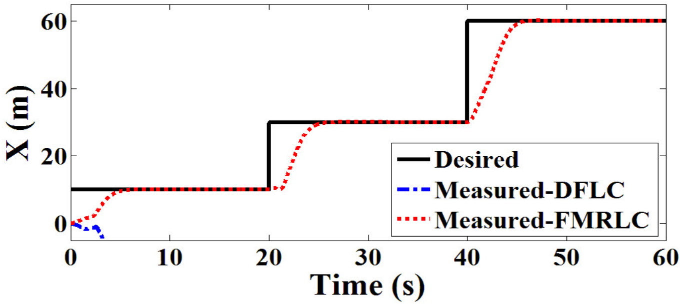
Study of the effect of changing the operating region is done as following: The operating point for X, Y, and Z is changed from 0 m to 60 m, also for θ_1 and θ_2 is changed from 0 to π rad, and finally, ψ is kept at 0 rad. The simulation results presented in Fig. 8 show that the FMRLC succeeds with satisfied accuracy and the DFLC fails to track because it need to retune its scaling factors.

VII. Conclusion

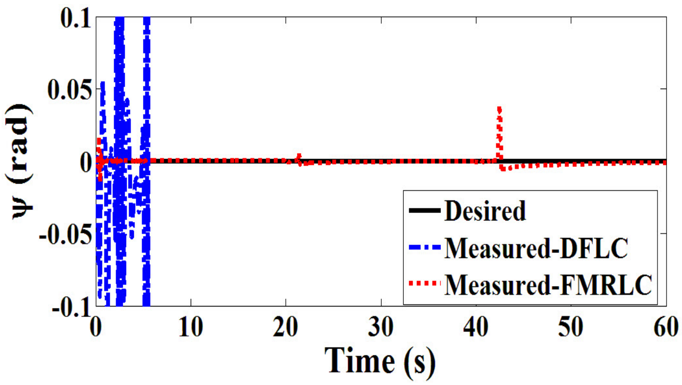
A new aerial manipulation system called ”Quadrotor–Manipulator System” was briefly described. Three control techniques were presented, feedback linearization based PID control, DFLC, and FMRLC. These controllers are tested to provide system stability and trajectory tracking under the effect of picking and placing a payload and the effect of changing the operating region. Simulation results shows that the feedback linearization technique fails to ensure system stability. In contrast, both DFLC and FMRLC can provide the system stability and good trajectory tracking under the effect of the payload but unlike FMRLC, the DFLC provides a steady state error. A comparison study is done between DFLC and FMRLC when changing the operating region and the results show the success of FMRLC and the failure of DFLC which causes system instability.

Acknowledgment

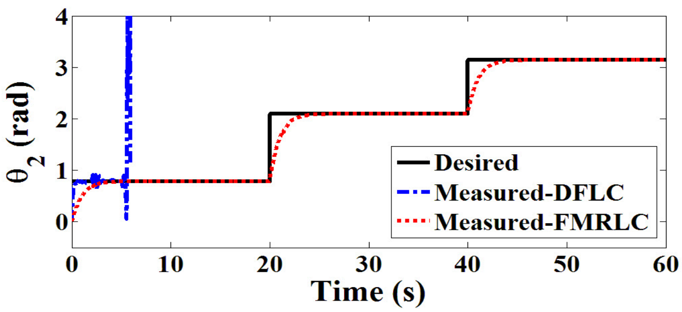
The first author is supported by a scholarship from the Mission Department, Ministry of Higher Education of the Government of Egypt which is gratefully acknowledged.



(a) X



(b) ψ



(c) θ_2

Fig. 7. Simulation Results for FBL Controller, DFLC, and FMRLC Under Effect of Adding/Releasing Payload: a) X, b) ψ, and c) θ_2.

REFERENCES

[1] D. Mellinger, Q. Lindsey, M. Shomin, and V. Kumar, "Design, modeling, estimation and control for aerial grasping and manipulation," in *Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on*, pp. 2668–2673, IEEE, 2011.

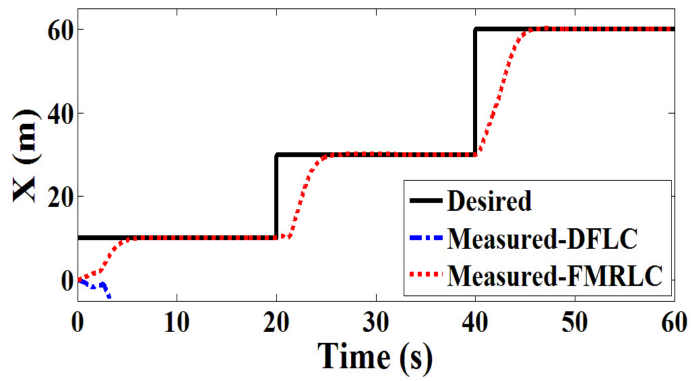
[2] M. Imran Rashid and S. Akhtar, "Adaptive control of a quadrotor with unknown model parameters," in *Applied Sciences and Technology (IBCAST), 2012 9th International Bhurban Conference on*, pp. 8–14, IEEE, 2012.

[3] J. Kim, M.-S. Kang, and S. Park, "Accurate modeling and robust hovering control for a quad-rotor vtol aircraft," in *Selected papers from the 2nd International Symposium on UAVs, Reno, Nevada, USA June 8–10, 2009*, pp. 9–26, Springer, 2010.

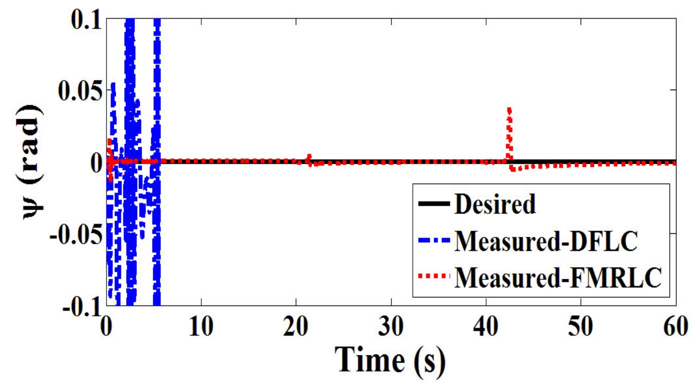
[4] H. Bouadi and M. Tadjine, "Nonlinear observer design and sliding mode control of four rotors helicopter," *Vorld Academy of Science, Engineering and Technology*, vol. 25, pp. 225–229, 2007.

[5] T. Bresciani, "Modelling, identification and control of a quadrotor helicopter," Master's thesis, Lund University, Lund, Sweden, 2008.

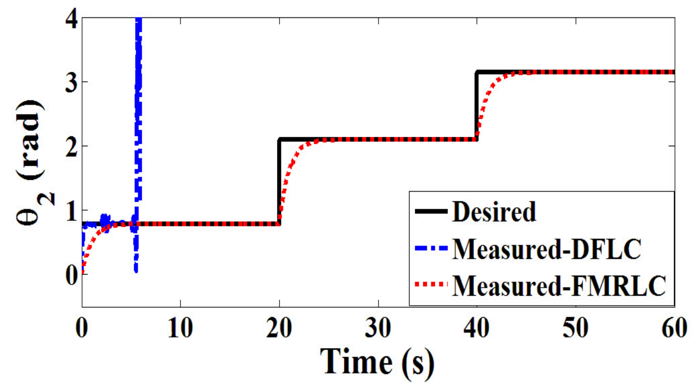
[6] M. Elsamanty, M. Fanni, and A. Ramadan, "Novel hybrid ground/aerial autonomous robot," in *Innovative Engineering Systems, 012 IEEE/RAS International Conference on*, pp. 103–108, IEEE, 2012.



(a) X



(b) ψ



(c) θ_2

Fig. 8. Simulation Results for Studying Effect of Changing the Operating Region in Case of FMRLC and DFLC: a) X, b) ψ, and c) θ_2.

[7] A. Khalifa, M. Fanni, A. Ramadan, and A. Abo-Ismail, "Modeling and control of a new quadrotor manipulation system," in *Innovative Engineering Systems, 2012 IEEE/RAS International Conference on*, pp. 109–114, IEEE, 2012.

[8] M. W. Spong, S. Hutchinson, and M. Vidyasagar, *Robot modeling and control*. John Wiley & Sons New York, 2006.

[9] L.-W. Tsai, *Robot analysis: the mechanics of serial and parallel manipulators*. Wiley-Interscience, 1999.

[10] K. M. Passino and S. Yurkovich, *Fuzzy control*. Citeseer, 1998.

[11] Z. Kovačić and S. Bogdan, *Fuzzy controller design: theory and applications*, vol. 19. CRC Press, 2006.

[12] S. A. Raza and W. Gueaieb, "Intelligent flight control of an autonomous quadrotor," *InTech*, 2010.

[13] J. R. Layne and K. M. Passino, "Fuzzy model reference learning control," in *Control Applications, 1992., First IEEE Conference on*,

pp. 686–691, IEEE, 1992.

[14] W. A. Kwong and K. M. Passino, “Dynamically focused fuzzy learning control,” *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, vol. 26, no. 1, pp. 53–74, 1996.