Proceedings of the SICE Annual Conference 2016
Tsukuba, Japan, September 20-23, 2016

On the Fault Tolerant Control of a Quadrotor Manipulation System via MPC and DOb Approaches

Ahmed Khalifa[1], Mohamed Fanni[2] and Toru Namerikawa[3†]

[1]Department of System Design Engineering, Keio University, Yokohama, Japan. On study leave from Department of Mechatronics and Robotics Engineering, Egypt-Japan University of Science and Technology, Alexandria, Egypt.
(E-mail: ahmed.khalifa@ejust.edu.eg)
[2]Department of Mechatronics and Robotics Engineering, Egypt-Japan University of Science and Technology. On leave from Department of Production Engineering and Mechanical Design, Mansoura University, Mansoura, Egypt.
(E-mail: mohamed.fanni@ejust.edu.eg)
[3]Department of System Design Engineering, Keio University, Yokohama, Japan.
(E-mail: namerikawa@nl.sd.keio.ac.jp)

Abstract: Recently, the Aerial Manipulation System becomes very attractive for a wide range of applications due to its unique features. However, control of such system is quite challenging. One of the critical challenge is that this system is very susceptible to actuators' faults. In this paper, a Passive Fault Tolerant Control System is proposed to address this issue with robust and optimal performance. The robustness is achieved using a linear Disturbance Observer (DOb) loop. Based on the linearization capability of DOb, a standard Model Predictive Control (MPC) is then used and the resulting control scheme is characterized by both a low computational load and optimal actuators' efforts with respect to conventional nonlinear robust solutions. This controller is tested to achieve the tracking of a point-to-point task space references under the effect of actuators' faults, picking/releasing a payload, changing the operating region, and measurement noise. Efficacy of the proposed technique is verified via numerical simulations.

Keywords: Aerial Manipulators, Fault Tolerant Control, Disturbance Observer, Model Predictive Control.

1. INTRODUCTION

Recently, aerial manipulators have much interest because they have several vital applications in the places which are not accessible by ground robots. Due to the superior mobility of quadrotors, they are utilized for mobile manipulation. Several researches have been introduced in the area of aerial manipulation [1–4]. However, the previous introduced systems in the literature that use a gripper suffer from the limited allowable DOF of the end-effector. The other systems have a manipulator with either two DOF but in certain topology that disables the end-effector to track arbitrary 6-DOF trajectory, or more than two DOF which decreases greatly the possible payload carried by the system. Moreover, the control schemes that were presented in the literature for such robots are based on nonlinear controllers which are very complicated and have very high computational cost.

In [5], we propose a new aerial manipulation system that consists of a 2-link manipulator attached to the bottom of a quadrotor. This new system presents a solution for the limitations found in the current quadrotor manipulation systems. Firstly, our proposed aerial manipulator has the capability of manipulating the objects with arbitrary location and orientation because it posses 6-DOF. Secondly, it is based on a minimum manipulator weight for aerial manipulation (2-DOF manipulator). Thirdly, the manipulator provides sufficient and controlled distance between quadrotor and object location.

There are several issues when working with the aerial manipulation systems. The first one is achieving the position holding. In order to achieve this task, a robust control must be implemented such that it can cope up with the uncertainties and disturbances, faulty situations of the actuators, payload changes, operation region changes, and measurement noise. The second issue with this system is the limitations of the actuators (thrusters and manipulator motors) which may degrade the performance if it is not considered during controller design. The third issue is the speed of the controller response which must be fast enough to be suitable with the high speed dynamics of the flying robot.

The robustness and optimal performance issues are solved in [6], in which, we propose a DOb based control technique to achieve the required robustness. The DOb estimate the nonlinear terms and uncertainties then compensates them such that the robotic system acts like a multi-SISO linear systems. Therefore, it is possible to rely on the standard MPC methodology to design the controller of the outer loop such that the system performance can be adjusted to achieve desired tracking accuracy and speed, and actuators constraints with low computational load and optimal control effort.

The Fault Tolerant Control System (FTCS) is a control system with the ability to tolerate faults automatically and to achieve the desired performance under the fault conditions in the system components (e.g. Actuators and Sensors). There are two different types of FTCS; one of them is the Passive Fault Control System (PFTCS), and the other is the Active Fault Control System (AFTCS). PFTCS is designed based on a fixed controller which should be robust enough to withstand the possible malfunctions during system operation. AFTCS

† Toru Namerikawa is the presenter of this paper.

978-4-907764-50-0 PR0001/16 ¥400 ©2016 SICE

is designed based on a reconfigurable controller with a Fault Detection and Diagnosis scheme to provide on-line system monitoring. Unlike the AFTCS, PFTCS does not require a reconfiguration control law and a Fault Detection and Diagnosis scheme. In this paper, the passive fault tolerant capability of the MPC/DOb approach against the actuators' faults is investigated.

This paper is organized as follows: In section 2, the considered robotic system is described, and kinematic and dynamic analysis are reviewed. The control problem to solve is formulated and described in section 3. In section 4, simulation results using MATLAB/SIMULINK are presented. Finally, the main contributions are concluded in section 5.

2. SYSTEM MODELING

Design and modeling of the quadrotor manipulation system are presented in details in [5]. The system consists mainly of two parts; the quadrotor and the manipulator. 3D CAD model of it is shown in Fig. 1. Fig. 2 presents a

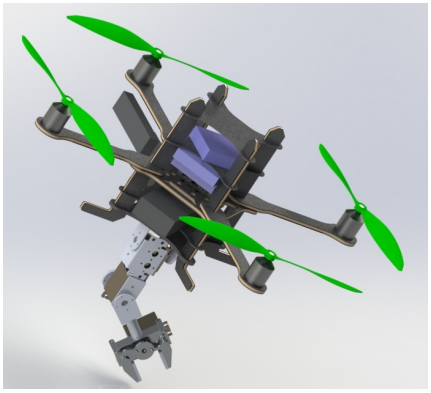


Fig. 1: 3D CAD model of the Quadrotor Manipulation System

sketch of such system with the relevant frames which indicates the unique topology that permits the end-effector to achieve arbitrary pose. The frames satisfy the Denavit-Hartenberg (DH) convention. The quadrotor components

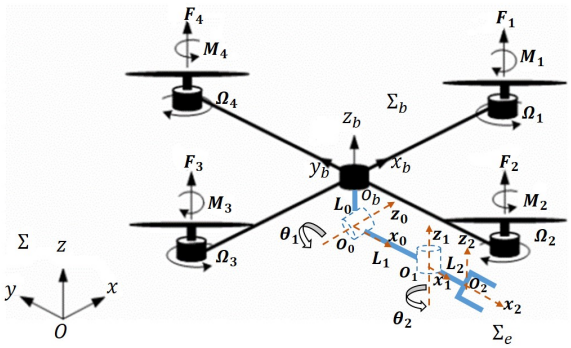


Fig. 2: Schematic diagram of Quadrotor Manipulation System with relevant frames

are selected such that it can carry a payload equals $500g$ (larger than the total arm weight and the maximum payload). Asctec pelican quadrotor is used as the quadrotor platform. The maximum thrust force for each rotor is $8N$ as obtained from an identification process.

The arm components are designed, selected, purchased and assembled such that the total weight of the arm is $200g$, has maximum reach in the range between $22cm$ to $25cm$, and can carry a payload of $200g$. Three DC motors, (HS-422 (Max torque = $0.4N.m$) for gripper, HS-5485HB (Max torque = $0.7N.m$) for joint 1, and HS-422 (Max torque = $0.4N.m$) for joint 2), are used.

Each rotor j has angular velocity Ω_j and it produces thrust force F_j and drag moment M_j which are given by

$$F_j = K_{f_j}\Omega_j^2, \tag{1}$$

$$M_j = K_{m_j}\Omega_j^2, \tag{2}$$

where K_{f_j} and K_{m_j} are the thrust and drag coefficients. These coefficients will be utilized later to simulate the effects of the actuators' faults.

Let Σ_b, O_b- x_b y_b z_b, denotes the vehicle body-fixed reference frame with origin at the quadrotor center of mass, see Fig. 2. Its position with respect to the world-fixed inertial reference frame, Σ, O- x y z, is given by the (3×1) vector $p_b = [x, y, z]^T$, while its orientation is represented by Φ_b=$[\psi, \theta, \phi]^T$, and is given by a rotation matrix R_b.

Let us consider the frame Σ_e, O_2- x_2 y_2 z_2, attached to the end-effector of the manipulator, see Fig. 2. Thus, the position of Σ_e with respect to Σ is given by

$$p_e = p_b + R_b p_{eb}^b, \tag{3}$$

where the vector p_{eb}^b describes the position of Σ_e with respect to Σ_b expressed in Σ_b. The orientation of Σ_e can be defined by the rotation matrix

$$R_e = R_b R_e^b, \tag{4}$$

where R_e^b describes the orientation of Σ_e w.r.t Σ_b.

The dynamical model of the quadrotor-manipulator system can be written as follows:

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) + d_{ex} = \tau, \qquad \tau = Bu, \tag{5}$$

where $q = [x, y, z, \psi, \theta, \phi, \theta_1, \theta_2]^T \in R^8$ is the generalized coordinates, $M \in R^{8\times8}$ represents the symmetric and positive definite inertia matrix of the combined system, $C \in R^8$ is the matrix of Coriolis and centrifugal terms, $G \in R^8$ is the vector of gravity terms, d_{ex} $\in R^8$ is vector of the external disturbances, $\tau \in R^{8\times8}$ is vector of the generalized input torques/forces, $u = [F_1, F_2, F_3, F_4, \tau_{m_1}, \tau_{m_2}]^T \in R^6$ is vector of the actuators' inputs, $B = HN \in R^{8\times6}$ is the input matrix which is used to generate the body forces and moments from the actuators' inputs. $N \in R^{8\times6}$ is given by

$$N = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 1 & 1 & 1 & 0 & 0 \\ \gamma_1 & -\gamma_2 & \gamma_3 & -\gamma_4 & 0 & 0 \\ -d & 0 & d & 0 & 0 & 0 \\ 0 & -d & 0 & d & 0 & 0 \\ 0 & 0 & 0 & 0 & K_{\tau_1} & 0 \\ 0 & 0 & 0 & 0 & 0 & K_{\tau_2} \end{bmatrix}, \tag{6}$$

where $\gamma_j = K_{m_j}/K_{f_j}$, K_{τ_1} and K_{τ_2} are the motor's constants of joints 1 and 2, respectively, and $H \in R^{8\times 8}$ transforms body input forces to be expressed in Σ and is given by

$$H = \begin{bmatrix} R_b & O_3 & O_2 \\ O_3 & T_b^T R_b & O_2 \\ O_{2x3} & O_{2x3} & I_2 \end{bmatrix}. \quad (7)$$

3. CONTROLLER DESIGN

This section presents the proposed motion control strategy. In this control strategy, the system nonlinearities, uncertainties, external disturbances, τ^{dis}, are treated as disturbances which will be estimated, $\hat{\tau}^{dis}$, and canceled by the DOb in the inner loop. The system can be now considered as linear SISO plants and thus the MPC is used in the external loop to achieved the objective performance for the system by producing τ^{des}.

3.1. Disturbance Observer Loop

A block diagram of the DOb inner loop is shown in Fig. 3. In this figure, $M_n \in R^{8\times 8}$ is the system nominal inertia matrix, τ and τ^{des} are the robot and desired inputs, respectively, $P = diag([g_1, ..., g_i, ..., g_8])$ with g_i is the bandwidth of the i^{th} variable of q, $Q(s) = diag([\frac{g_1}{s+g_1}, ..., \frac{g_i}{s+g_i}, ..., \frac{g_8}{s+g_8}]) \in R^{8\times 8}$ is the matrix of the low pass filter of DOb. It is a well known fact that a DOb requires precise velocity measurement. Therefore, the velocity is estimated by using a low pass filter, $Q_v(s) = diag([\frac{g_{v_1}}{s+g_{v_1}}, ..., \frac{g_{v_i}}{s+g_{v_i}}, ..., \frac{g_{v_8}}{s+g_{v_8}}]) \in R^{8\times 8}$, and with cut-off frequency of $P_v = diag([g_{v_1}, ..., g_{v_i}, ..., g_{v_8}])$ (i.e. precise velocity measurement is achieved in a predetermined bandwidth). τ^{dis} represents the system disturbances including the Coriolis, centrifugal and gravitational terms, and $\hat{\tau}^{dis}$ represents the system estimated disturbances.

The system disturbance τ^{dis} can be assumed as:

$$\begin{aligned} \tau^{dis} &= (M(q) - M_n)\ddot{q} + \tau^d, \\ \tau^d &= C(q,\dot{q})\dot{q} + G(q) + d_{ex}. \end{aligned} \quad (8)$$

Substituting from (8), then (5) can be rewritten as

$$M_n\ddot{q} + \tau^{dis} = \tau. \quad (9)$$

Now if the disturbance observer performs optimally, that is $\hat{\tau}^{dis} = \tau^{dis}$, the dynamics from the DOb loop input τ^{des} to the output of the robot manipulator is given by

$$M_n\ddot{q} = \tau^{des}. \quad (10)$$

Since M_n is a diagonal matrix, the system can be treated as multi-decoupled linear SISO systems as following

$$M_{n_{ii}}\ddot{q}_i = \tau_i^{des}, \quad (11)$$

or simply in the acceleration space as

$$\ddot{q}_i = \ddot{q}_i^{des}. \quad (12)$$

3.2. Model Predictive Control

By virtue of the rejection of the nonlinearities, disturbances, uncertainties, and noise through the use of DOb internal loop, the MPC controller have not to consider uncertainties and can designed based on the nominal 8 SISO decoupled models of the system (12) with satisfying the actuators constraints, $u_{min} \le u \le u_{max}$.

The SISO model of the system is given by

$$\begin{aligned} \dot{x}_p(t) &= A_p x_p(t) + B_p u_{mpc}(t), \\ q_i(t) &= C_p x_p(t), \end{aligned} \quad (13)$$

where $x_p = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = \begin{bmatrix} \dot{q}_i \\ q_i \end{bmatrix}$, $A_p = \begin{bmatrix} 0 & 0 \\ 1 & 0 \end{bmatrix}$, $B_p = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$, $C_p = \begin{bmatrix} 0 & 1 \end{bmatrix}$, and $u_{mpc}(t)$ is the output of the MPC which will be $\ddot{q}^{des}$. The general design philosophy of model predictive control is to compute a trajectory of a future manipulated variable $u_{mpc}(t)$ to optimize the future behavior of the plant output $q_i(t)$. The optimization is performed within a limited time window starts by t_i with length T_p. To achieve offset free MPC, it will be designed based on an augmented model of the plant which is given as

$$\begin{aligned} \dot{x}(t) &= Ax(t) + B\dot{u}_{mpc}(t), \\ q_i(t) &= Cx(t), \end{aligned} \quad (14)$$

where $x(t) = \begin{bmatrix} \dot{x}_p(t) \\ q_i(t) \end{bmatrix}$, $A = \begin{bmatrix} A_p & O_{2\times 1} \\ C_p & 1 \end{bmatrix}$, $B = \begin{bmatrix} B_p \\ 0 \end{bmatrix}$, and $C = \begin{bmatrix} O_{1\times 2} & 1 \end{bmatrix}$.

The MPC design is based on the solution of the so-called Finite Horizon Optimal Control Problem which consists of minimizing a suitably defined cost function with respect to the control sequence. The cost function is given as:

$$\begin{aligned} J = \int_0^{T_p} & (q_r(t_i) - q(t_i + \iota|t_i))^T Q_{mpc}(q_r(t_i) \\ & - q(t_i + \iota|t_i)) + \dot{u}_{mpc}^T(\iota) R_{mpc} \dot{u}_{mpc}(\iota) d\iota, \end{aligned} \quad (15)$$

where Q_{mpc} and R_{mpc} are weighting matrices that must be positive definite to ensure the stability of the outer loop.

In our design, the controller models the system response using a generic function series approximation

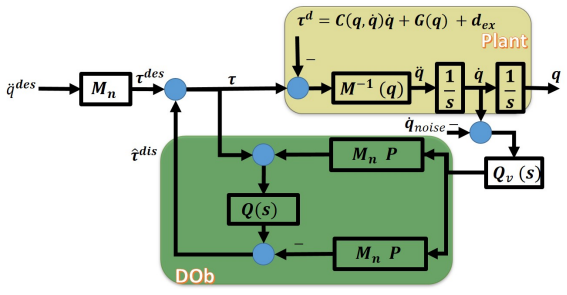


Fig. 3: Block diagram of DOb internal loop

technique based on Laguerre polynomials with parameters; a that is the scaling factor and b is the number of terms. By applying the principle of receding horizon control (i.e., the control action will use only the derivative of the future control signal at $\iota = 0$), one can get the derivative of the optimal control for the unconstrained problem.

To ensure the outer loop stability, an exponential weighted cost function [7] is used in which the system matrix, A, is modified to be $(A - \varrho I)$ and Q_{mpc} to be $(Q_{mpc} + 2\varrho P_{mpc})$, where P_{mpc} is the solution of the Riccati equation. The selection of weighting factor, ϱ, is to make sure that the design model with $(A - \varrho I)$ is stable with all eigen values on the left-half of the complex plane.

In order to specify the closed-loop response speed, a new tuning parameter, β, is used. During solution of Riccati equation, the A matrix is modified to $(A + \beta I)$ and Q_{mpc} to $(Q_{mpc} + 2(\varrho + \beta)P_{mpc})$.

Fig. 4 shows the complete block diagram of the proposed control system. Since the vehicle (quadrotor) is an under-actuated system, i.e., only 4 independent control inputs are available against the 6 DOF, the position and the yaw angle are usually the controlled variables, while pitch and roll angles are used as intermediate control inputs for horizontal positions control. Therefore, the proposed control system consists from two DOb-based controllers by dividing q in to two parts; one for $\zeta = [x, y, z, \psi, \theta_1, \theta_2]^T$ (with MPC_ζ, M_{n_ζ}, P_ζ, Q_ζ) and the other for $\sigma_b = [\theta, \phi]^T$ (with MPC_σ, M_{n_σ}, P_σ, Q_σ). The desired end-effector's pose, $\chi_{e,r}$, is fed to a point-to-point inverse kinematics algorithm [8] such that the desired vehicle/joint space trajectories, $\zeta_r(t)$, are obtained. After that, the controller block receives the desired trajectories and the feedback signals from the system and provides the control signal, τ. The desired values for the intermediate controller, $\sigma_{b,r}$, are obtained from the output of position controller, τ_ζ, through the following relation

$$\sigma_{b,r} = \frac{1}{\tau_\zeta(3)} \begin{bmatrix} C(\psi) & S(\psi) \\ S(\psi) & -C(\psi) \end{bmatrix} \begin{bmatrix} \tau_\zeta(1) \\ \tau_\zeta(2) \end{bmatrix}. \tag{16}$$

Note that $C(.)$ and $S(.)$ are short notations for $\cos(.)$ and $\sin(.)$, respectively. The output of the two controllers, τ_ζ and τ_σ, are mixed to generate the final control vector, τ, which is then converted to the forces/torques applied to quadrotor/manipulator through the following relation

$$u = B_6^{-1} \begin{bmatrix} \tau_\zeta(1:4) \\ \tau_\sigma \\ \tau_\zeta(5:6) \end{bmatrix}, \tag{17}$$

where $B_6 \in R^{6\times 6}$ is part of B matrix and it is given by $B_6 = B(3:8, 1:6)$.

3.3. Fault Tolerant Control Design

The control objective is to design a PFTCS based on the MPC/DOb control approaches such that the tracking error can be minimized even if the actuators are faulty. By utilizing the control matrix, N, several actuator fault scenarios can be simulated and evaluated. This can be implemented by varying the actuators parameters ($\gamma_j = K_{m_j}/K_{f_j}$, K_{τ_1} and K_{τ_2}) in (17).

4. SIMULATION RESULTS

In this section the previously proposed control system is simulated using MATLAB/SIMULINK program to the model of the considered aerial manipulation robot. Note that the model has been identified on the basis of real data through experimental tests, and a normally distributed measurement noise, with mean of 10^{-3} and standard deviation of 5×10^{-3}, has been added to the measured signals. In addition, the external disturbances are simulated as picking a payload of value 150 g at instant 15 s and placing it at instant $55s$. As a result, the simulation environment is quite realistic. The identified parameters are given in [6].

To achieve task space control, the desired values of end-effector pose are used to generate the desired trajectories for ζ using the inverse kinematics and then applied to the algorithm for generating the desired trajectories in the quadrotor/joint space which will provide Quintic polynomial trajectories as the reference trajectories. By using these types of trajectories for the joint space control, one can avoid the vibrational modes because they have sinusoidal acceleration. Parameters of DOb-based control is given in Table 1.

4.1. Fault Simulation

Simulation of the fault scenarios is presented in Fig. 5. The actuators' losses with different percentages and at different instants are simulated and evaluated.

The simulation results in the quadrotor/joint space are presented in Fig. 6. These results show that the proposed PFCS has the capability to handle the faults even if the actuators lose their efficiency simultaneously with a fixed controller parameters (i.e. There is no need to change the parameters of the controller.). Fig. 7 shows the response of system in the task space (the actual end-effector position and orientation can be found from the forward kinematics). Fig. 8 shows the control effort (the required thrust forces and manipulator torques), which ensure that

Table 1: Controller parameters

Par.	Val.	Par.	Val.
M_{n_ζ}	$diag\{1.2, 1.2, 2, 0.5, 0.005, 0.005\}$	ϱ_{mpc_ζ}	10
M_{n_σ}	$diag\{0.5, 0.5\}$	Q_{mpc_ζ}	C^TC
u_{max}	$[6, 6, 6, 6, 0.7, 0.4]^T$	Q_{mpc_σ}	C^TC
u_{min}	$[0, 0, 0, 0, -0.7, -0.4]^T$	R_{mpc_ζ}	I_6
P_ζ	$diag\{2.4, 2.4, 14.5, 0.6950, 19.5, 4.5\}$	R_{mpc_σ}	I_2
P_σ	$diag\{1.3, 0.76\}$	ϱ_{mpc_σ}	15
β_{mpc_ζ}	5	T_p	70
β_{mpc_σ}	10	b	6
g_{v_i}	100	a	0.6

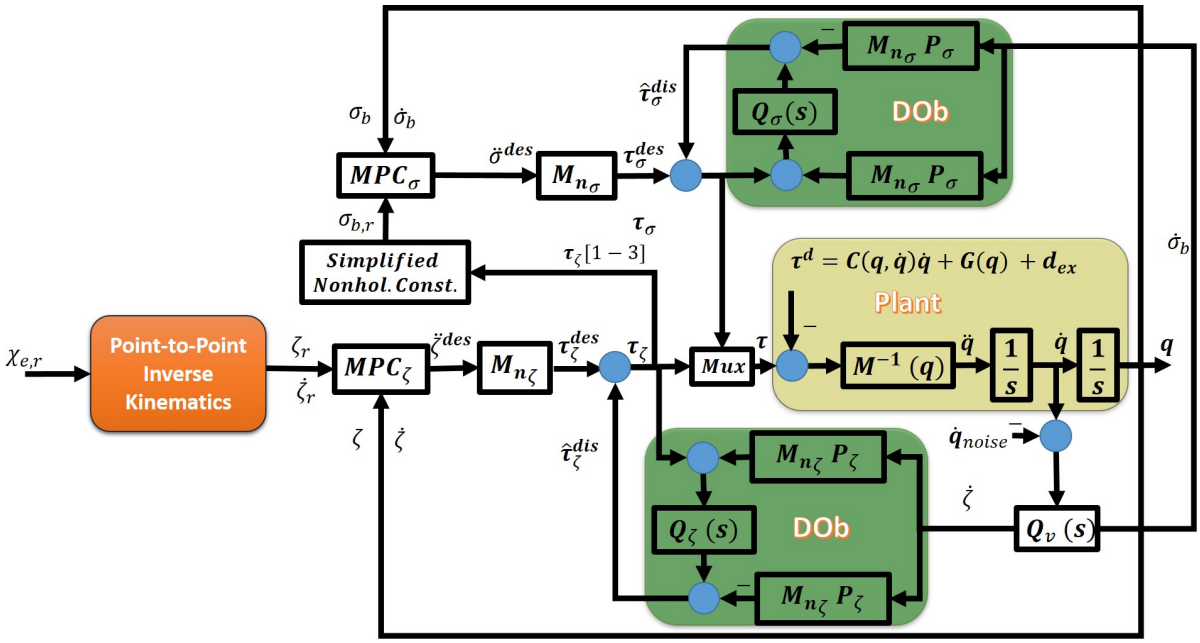


Fig. 4: Block diagram of the detailed control system for the quadrotor manipulation system

they are in the allowable limits. Therefore, one can contend that the proposed motion control scheme is able to achieve the control objectives.

5. CONCLUSION

The problem of the fault tolerant control of an aerial manipulation robot is presented. A Passive Fault Tolerant Control System is designed based on the DOb and MPC principles. The DOb loop is used to enforce robust linear input/output behavior of the plant by canceling the effect of disturbances, faulty actuators, measurement noise, and plant/model mismatch. After that, the MPC controller is used in the external loop to achieve the required closed loop performance and control objectives with low computational load. The system is simulated using MATLAB/SIMULINK. Simulation results enlighten the efficiency of the proposed controller.

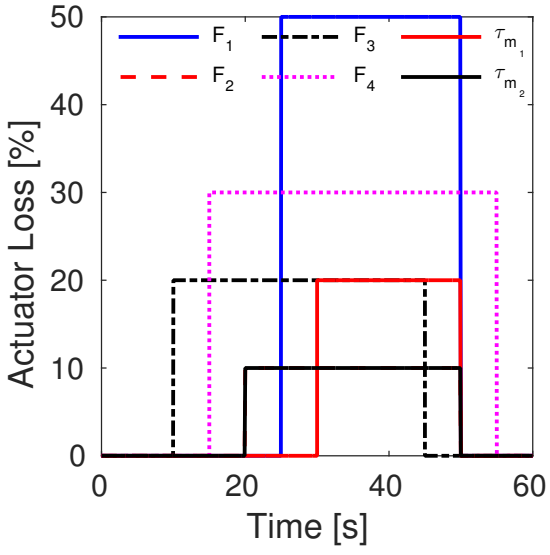


Fig. 5: Actuators Loss

REFERENCES

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

[2] N. Michael, J. Fink, and V. Kumar, "Cooperative manipulation and transportation with aerial robots," *Autonomous Robots*, vol. 30, no. 1, pp. 73–86, 2011.

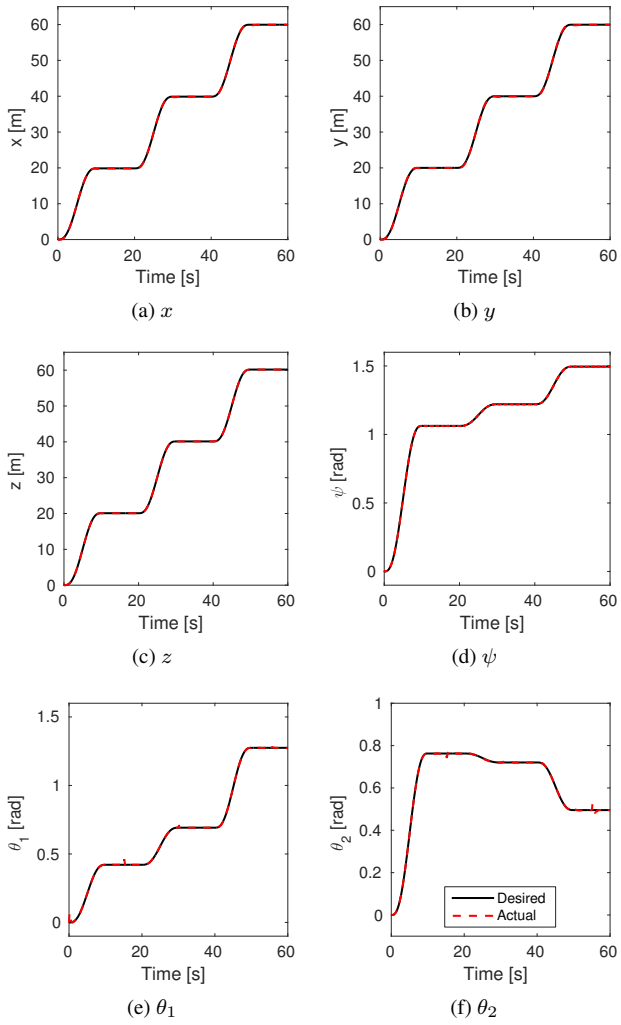
[3] C. M. Korpela, T. W. Danko, and P. Y. Oh, "Mm-uav: Mobile manipulating unmanned aerial vehicle," *Journal of Intelligent & Robotic Systems*, vol. 65, no. 1-4, pp. 93–101, 2012.

[4] G. Heredia, A. Jimenez-Cano, I. Sanchez, D. Llorente, V. Vega, J. Braga, J. Acosta, and A. Ollero, "Control of a multirotor outdoor aerial manipulator," in *Intelligent Robots and Systems (IROS 2014), 2014 IEEE/RSJ International Conference on*. IEEE, 2014, pp. 3417–3422.

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

[6] A. Khalifa, M. Fanni, and T. Namerikawa, "Mpc and dob-based robust optimal control of a new quadrotor manipulation system," in *European Control Conference, Denmark*. IEEE, 2016, p. (Accepted).

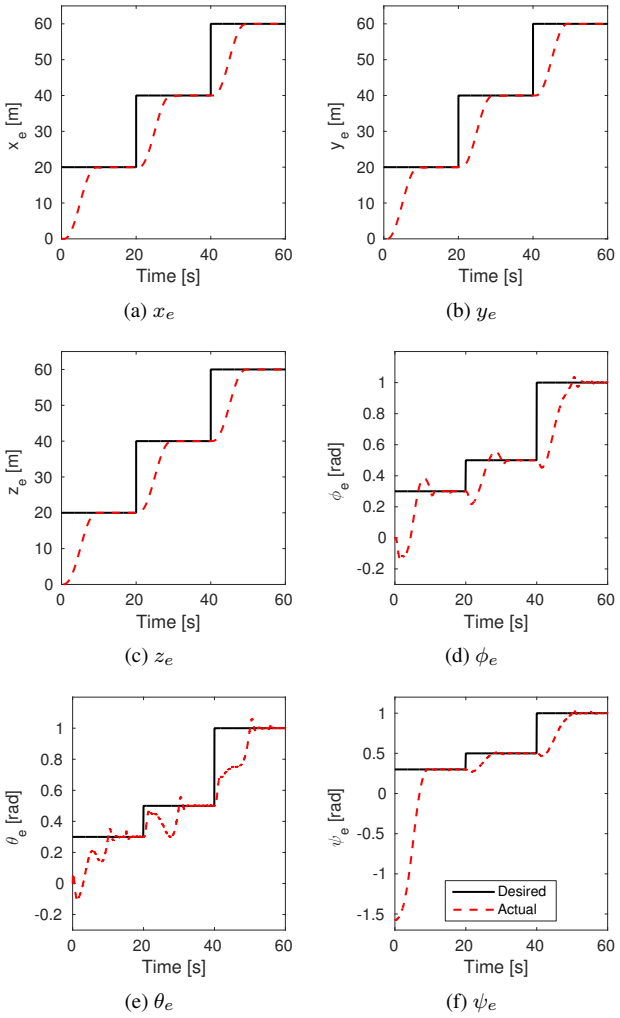
[7] L. Wang, *Model predictive control system design and implementation using MATLAB®*. Springer Science & Business Media, 2009.



(a) x (b) y (c) z (d) ψ (e) θ_1 (f) θ_2

Fig. 6: The actual response of the Quadrotor/ Manipulator variables: a) x, b) y, c) z, d) ψ, e) θ_1, and f) θ_2

[8] A. Khalifa and M. Fanni, "Position analysis and control of a new quadrotor manipulation system," *International Journal of Robotics and Automation*, 2016.



(a) x_e (b) y_e (c) z_e (d) ϕ_e (e) θ_e (f) ψ_e

Fig. 7: The actual response of the end-effector position and orientation: a) x_e, b) y_e, c) z_e, d) ϕ_e, e) θ_e, and f) ψ_e

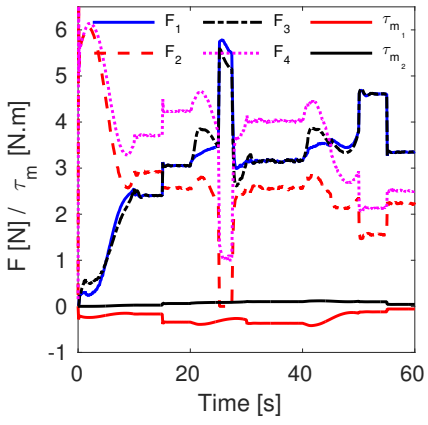


Fig. 8: The required controller efforts