2014 International Conference on Unmanned Aircraft Systems (ICUAS)
May 27-30, 2014. Orlando, FL, USA

Model-Free Control of a Quadrotor Vehicle

Younes Al Younes, Ahmad Drak, Hassan Noura, Abdelhamid Rabhi and Ahmed El Hajjaji

Abstract— **In this paper, the Model-Free Control (MFC) is used and tested on a Multi-Input-Multi-Output (MIMO) nonlinear system, which is a quadrotor vehicle. The system is decomposed into different-dependent Single-Input-Single-Output (SISO) sub-systems, where the MFC algorithm is applied on each one of them. MFC aims at compensating the time-varying disturbances and un-modeled system dynamics that the optimal feedback controller fails to cope with. A comparison between LQR feedback controller with and without the MFC will be implemented and tested on a real quadrotor vehicle.**

Different flight test results validate the importance of using the MFC, and its capability to control the quadrotor system that is using a degraded feedback controller, as well.

I. Introduction

The quadrotor vehicle is a type of the Vertical-Take-Off-and-Landing (VTOL) rotorcraft. It has four rotors; each generates a lifting force due to the rotation of the propellers. Two rotors rotate clockwise and two counter-clockwise. The difference in the thrust between the four rotors gives the vehicle the ability to move in six-Degree-Of-Freedom (6-DOF), which are the roll, pitch, yaw angles and the position x, y, z.

Recently, several countries are studying the feasibility of deploying quadrotors for the application of food and parcel delivery. The quadrotors are relatively cheap and easy to fly, thus making them the best choice when it comes to testing different control strategies on a UAV. Control strategies can be classified mainly into two different categories, linear and non-linear control. A great deal of linear and non-linear control strategies have been implemented on quadrotors [1-6].

The Linearization of a highly nonlinear model degrades the controlling performance, and in such a situation, the linear control algorithms fail to control the vehicle at points rather than the operating point. This pushes toward developing alternative methodologies to control the nonlinear model of the quadrotor while maneuvering aggressively and flying in harsh environments.

The author of [2] applied the backstepping control technique to stabilize the attitude of an indoor micro quadrotor. The controller showed the ability to control the orientation angles in the presence of perturbations. Others used various nonlinear control techniques to control the quadrotor, such as: Hybrid Backstepping and Sliding-Mode control [7, 8].

In contrary, designing a robust nonlinear controller to deal with the nonlinearities of the model requires a complete knowledge of the nonlinear system dynamics, which is a worrying task for the researchers. In another word, extracting a perfect model to describe the exact motion of the system is almost impossible. Besides that, faults and disturbances could lead to dramatic changes in the system dynamics, and therefore to unwanted behavior in control.

Lately, Model-Free Control (MFC) is introduced by M. Fliess and C. Join [9]. MFC is a control technique based on local-model that is continuously updated according to the input-output behavior. It is designed to overcome the un-modeled dynamics and uncertainties of the system, especially if it's a nonlinear system.

MFC works side by side with the feedback controller. If the feedback controller is a classical PID controller, then the combination is known as intelligent-PID controller (iPID) [9]. On the one hand, a general knowledge about the system model is required to determine and tune the controller's parameters. On the other hand, the MFC will deal with the model uncertainties and external disturbances that could have crucial effect on the system stability. Therefore, the model-free terminology does not mean that the system is considered as a black-box, but a generic knowledge of the system dynamics is needed.

Research is being conducted on the effect of the MFC on Shape Memory Alloys (SMA) [10], DC-DC converters [11], active magnetic bearing [12] and planar manipulator [13].

The objective of this paper is to investigate the implementation of the Model-Free Controller (MFC) on a quadrotor UAV.

In section II, MFC algorithm will be presented. Section III will go through an overview about the quadrotor setup that is used to perform the control strategy. The real flight test results will be performed in section IV, and finally conclusions about this work will be presented in section V.

* Research supported by United Arab Emirates University, UAEU.

Younes Al Younes is Mechanical Engineering Faculty at Higher Colleges of Technology, Al Ain, UAE, and PhD student at the University of Picardie Jules Verne, Amiens, France (phone: 971-50-1685048; fax: 971-3-782-0099; e-mail: yalyounes@hct.ac.ae).

Ahmad Drak is Electrical Engineer and is currently a research assistant in the Unmanned Vehicles Lab at the United Arab Emirates University (UAEU), Al Ain, UAE (phone 971-50-3327411 email: ahmad.drak@gmail.com)

Hassan Noura, PhD, is Professor and Chairman of Electrical Engineering Department at United Arab Emirates University, Al Ain, UAE (e-mail: hnoura@uaeu.ac.ae).

Abdelhamid Rabhi, PhD, is Assistant Professor in Modeling, Information & System Lab - Control & Vehicle Group- at University of Picardie, Amiens, France (e-mail: abdelhamid.rabhi@u-picardie.fr).

Ahmed El Hajjaji, PhD, is Professor in Modeling, Information & System Lab - Control & Vehicle Group- at University of Picardie, Amiens, France (e-mail: ahmed.hajjaji@u-picardie.fr).

II. MODEL-FREE CONTROL (MFC)

Consider the following n[th] order nonlinear SISO system:

$$y^{(n)} = f\big(y, \dot{y}, \dots, y^{(n)}\big) + bu \qquad (1)$$

Where, $f(\cdot)$ is the modeled system dynamics, u is the system input, and b is unknown input factor.

The un-modeled dynamics and uncertainties with the unknown input factor could be represented in the system as $f_e(\cdot)$:

$$f_e(\cdot) = \text{Model Uncertainties } + (b - \beta)u \qquad (2)$$

Thus, the system could be written as follows:

$$y^{(v)} = f(\cdot) + f_e(\cdot) + \beta u \qquad (3)$$

Where v is representing the order of the anticipated model, and β is the estimate of the unknown scaling factor b that is going to be determined by the operator to ensure a certain control performance.[13]

The input-output relation could be represented by an ultra-local model that is continuously restructured:

$$y^{(v)} = F + \beta u \qquad (4)$$

Where, F is a continuously updated value that represents the overall time-varying dynamics of the system $\big(f(\cdot) + f_e(\cdot)\big)$, and it could be approximated to attenuate the noise produced from the derivative $y^{(v)}$. [14]

$$F = y^{(v)} - \beta u \qquad (5)$$

It is worth to mention that the value of estimation is valid for a short period of time and it should be continuously updated [14]. In general, the model-free control input could be written as follows:

$$u = -\frac{F - y_d^{(v)} + u_c}{\beta} \qquad (6)$$

Where, $y_d^{(v)}$ is the v^{th} derivative of the desired trajectory, and u_c is the control input of the feedback controller.

By substituting (6) in (5),

$$y^{(v)} = F + \beta\left(-\frac{F - y_d^{(v)} + u_c}{\beta}\right) = y_d^{(v)} - u_c \qquad (7)$$

This is will yield to,

$$y^{(v)} - y_d^{(v)} + u_c = 0 \qquad (8)$$

$$e^{(v)} + u_c = 0 \qquad (9)$$

Where, $e^{(v)}$ is the v^{th} derivative error of $(e = y - y_d)$, and u_c should be selected to lead to a linear differential equation that is asymptotically converged to a desired trajectory. [14]

The operator has to choose the proper value of v according to the system stability and the type of the feedback controller used in the system. For instant, iPID and iPD would use a second order system with $v = 2$, and both iPI and iP will suit a system with $v = 1$.

To make this statement clear, assume $v = 1$, then (9) will be written as:

$$\dot{e} + u_c = 0 \qquad (10)$$

A second order differential equation could be extracted, if P or PI controller is implemented:

$$\dot{e} + K_P e + K_I \int e = 0 \qquad (11)$$

Similarly, if $v = 2$, then a third order differential equation will expressed, if PD or PID controller is used:

$$\ddot{e} + K_D\, \dot{e} + K_P\, e + K_I \int e = 0 \qquad (12)$$

Fig. 1 depicts the general MFC scheme for a SISO system when $v = 2$.

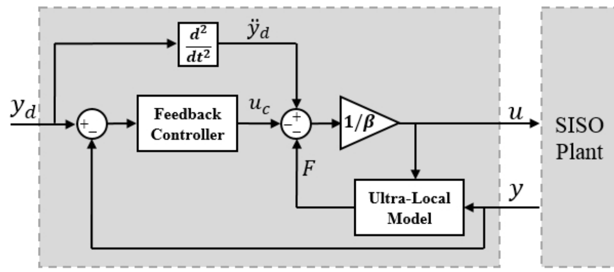

Figure 1. MFC scheme

As Fig. 1 shows, the ultra-local model in (4) will compute the value of F on every iteration of the closed loop controller. The updated value will estimate the system dynamics and inject a proper change in the control input of the SISO plant.

Within the ultra-local model block, the estimation of the derivative $\ddot{y}$ is done using different de-noising techniques [10-12, 14, 15]. For simplicity, low-pass filters are used in this project to attenuate the noisy signals produced from the numerical differentiation.

III. QUANSER PLATFORM

The QUANSER platform for quadrotor currently available in the United Arab Emirates University (UAEU). As shown in Fig. 2, the platform is mainly composed of a ground station (PC), Qball-X4 quadrotor, joystick and six Optitrack cameras. In addition to a safety net that surrounds the workspace to ensure the safety of the operator(s) and to avoid fatal crashes of the Qball-X4 as shown in Fig. 3.

The ground station has Matlab/Simulink and Quantum Architectures & Computations toolbox (QuArC). As a control-developing environment, QuArC is integrated with Simulink to develop and compile the proposed control algorithm [16]. The Optitrack cameras shown in Fig. 4 are used to localize the position of the Qball-X4.

The Qball-X4 is equipped with QUANSER Embedded Control Module (QECM), which is composed of a data acquisition card and the Gumstix QuArC target computer. The Gumstix allows rapid deployment of the control algorithm to the quadrotor. In addition to wireless module that allows a wireless communication to and from the ground station.

Figure 2. QUANSER's Qball-X4 UAV

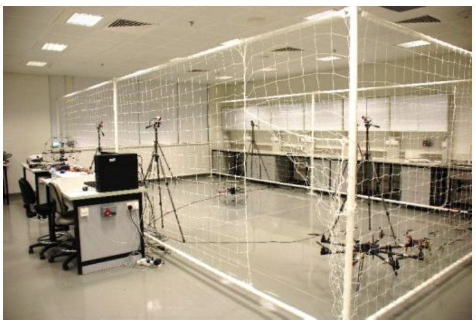

Figure 3. Safety net outlining the Qball-X4 workspace.

Figure 4. Optitrack camera used as a localization system.

The data acquisition card has high-resolution accelerometer, gyroscope, magnetometer IMU sensor and produces PWM signals to drive the motors. The downloaded controller run onboard the Qball-X4 and supports real-time sensor measurements and data logging, in addition to parameter tuning.

A. Qball-X4 Quadrotor Model

It is pertinent to discuss the actuator dynamics of the Qball-X4 system. A study about the system model had been provided in [17] and illustrated next.

The thrust generated by each propeller is modelled using the first-order equation:

$$F = K\,\frac{w}{s+w}\,u_r \tag{13}$$

The use of a state variable, q, will be needed to represent the actuator dynamics and is defined as follows:

$$q = \frac{w}{s+w}\,u_r \tag{14}$$

The actuator input signal is u_r, which is the PWM signal, while w is the actuator bandwidth and K is a positive gain.

The values of these parameters were calculated and verified by experimentation and can be seen in Table I.

The following state space representation describes the dynamics of the roll/pitch variables:

$$\begin{bmatrix}\dot{\theta}\\ \ddot{\theta}\\ \dot{q}\\ \dot{g}\end{bmatrix} = \begin{bmatrix}0 & 1 & 0 & 0\\ 0 & 0 & \frac{KL}{J_{roll/pitch}} & 0\\ 0 & 0 & -w & 0\\ 1 & 0 & 0 & 0\end{bmatrix}\begin{bmatrix}\theta\\ \dot{\theta}\\ q\\ g\end{bmatrix} + \begin{bmatrix}0\\ 0\\ w\\ 0\end{bmatrix}\Delta u \tag{15}$$

Where Δu is the difference in the inputs to the motors. The integrator $(\dot{g} = \theta)$ will be used in the feedback controller, where the fourth state is added to the state vector, as shown above in (15).

TABLE I. QBALL-X4 PARAMETERS

Parameter	Value
K	120 N
w	15 rad/s
J_{roll}	0.03 kg.m^2
J_{pitch}	0.03 kg.m^2
m	1.4 kg
K_y	4 N.m
J_{yaw}	0.04 kg.m^2
L	0.2 m

An assumption that roll and pitch angles are close to zero is used, this allows for the linearization of the height model dynamics equation to the following state space representation:

$$\begin{bmatrix}\dot{z}\\ \ddot{z}\\ \dot{q}\\ \dot{s}\end{bmatrix} = \begin{bmatrix}0 & 1 & 0 & 0\\ 0 & 0 & \frac{4K}{m} & 0\\ 0 & 0 & -w & 0\\ 1 & 0 & 0 & 0\end{bmatrix}\begin{bmatrix}Z\\ \dot{Z}\\ q\\ s\end{bmatrix} + \begin{bmatrix}0\\ 0\\ w\\ 0\end{bmatrix}u_t + \begin{bmatrix}0\\ -g\\ 0\\ 0\end{bmatrix} \tag{16}$$

Where, u_t is the total thrust from all motors. By applying the same assumption as above, the linear state space equations for the *x* and *y* positions are as follows:

$$\begin{bmatrix}\dot{x}\\ \ddot{x}\\ \dot{q}\\ \dot{s}\end{bmatrix} = \begin{bmatrix}0 & 1 & 0 & 0\\ 0 & 0 & \frac{4K}{m}\theta_{pitch} & 0\\ 0 & 0 & -w & 0\\ 1 & 0 & 0 & 0\end{bmatrix}\begin{bmatrix}X\\ \dot{X}\\ q\\ s\end{bmatrix} + \begin{bmatrix}0\\ 0\\ w\\ 0\end{bmatrix}\Delta u$$

$$\begin{bmatrix}\dot{y}\\ \ddot{y}\\ \dot{q}\\ \dot{s}\end{bmatrix} = \begin{bmatrix}0 & 1 & 0 & 0\\ 0 & 0 & \frac{-4K}{m}\theta_{roll} & 0\\ 0 & 0 & -w & 0\\ 1 & 0 & 0 & 0\end{bmatrix}\begin{bmatrix}Y\\ \dot{Y}\\ q\\ s\end{bmatrix} + \begin{bmatrix}0\\ 0\\ w\\ 0\end{bmatrix}\Delta u \tag{17}$$

The relationship between the torque generated by each motor and the PWM input signal can be expressed as:

$$\tau = K_y u_r \tag{18}$$

Where, K_y is a positive gain and its value can be seen in Table I. Then the yaw motion can be modeled as:

$$J_{yaw}\ddot{\psi} = \Delta\tau \tag{19}$$

Where, J_{yaw} is the rotational inertia about the z-axis, ψ is the yaw angel and $\Delta\tau = \tau_1 + \tau_2 - \tau_3 - \tau_4$. The yaw dynamics can be represented as follows:

$$\begin{bmatrix}\dot{\psi}\\ \ddot{\psi}\end{bmatrix} = \begin{bmatrix}0 & 1\\ 0 & 0\end{bmatrix}\begin{bmatrix}\psi\\ \dot{\psi}\end{bmatrix} + \begin{bmatrix}0\\ \frac{K_y}{J_{yaw}}\end{bmatrix}\Delta\tau \tag{20}$$

B. Quadrotor Control

QUANSER provides a ready-made controller to be downloaded onto the Qball-X4 upon first flight. The Linear-Quadratic Regulator (LQR) controller is mainly suitable for linear systems or linearized systems around a specific operating point. However, the tuning of the LQR control law proves to be challenging and highly dependent on the model of the system [18]. The LQR controller gains used in stabilizing the Qball-X4 orientation and position are presented in the supported documents by QUANSER [17].

The integral action and the derivative part are considered in computing the control inputs in the LQR controller that is used in the Qball-X4 quadrotor. This is will lead to the desired convergence of the error to zero as shown in equation (12), and therefore the MFC with $\nu = 2$ can be directly implemented to the existing LQR controller.

In this paper, the Qball-X4 quadrotor system will be decomposed into multi-SISO subsystems that are linked to each other. Six MFCs will be used each designated to control a certain state in the 6-DOF quadrotor system. The gain β for each controller will be identified empirically.

Three MFCs will be designed to control the altitude (z) and the planar position (x, y) of the quadrotor. Cascaded scheme of the controllers between the translational and rotational motion will be implemented. The output of the planar position controllers will feed two more MFCs for roll and pitch stabilization. In addition, another MFC will be designed for yaw control. The control architecture of the quadrotor system is depicted in Fig. 5.

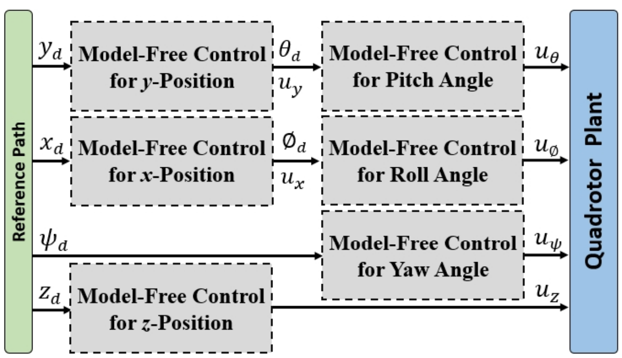

Figure 5. System control architicture.

IV. Flight Test Results

The objective of the experiment is to check the performance of the quadrotor in following a pre-generated square path using different control methods. The square path extends from -0.5 m to +0.5 m on the X axis and -0.5m to 0.5m on the Y axis. The Z axis (height) is kept constant at 1m. The quadrotor flies for 10 sec on each side of the square path, and stays 5 sec on each corner. The total real flight time from taking off till landing is 75 sec.

A. LQR Controller:

The system performance of the ready-made optimal controller (LQR) is depicted in Fig. 6. The quadrotor shows some oscillation and offset from desired trajectory.

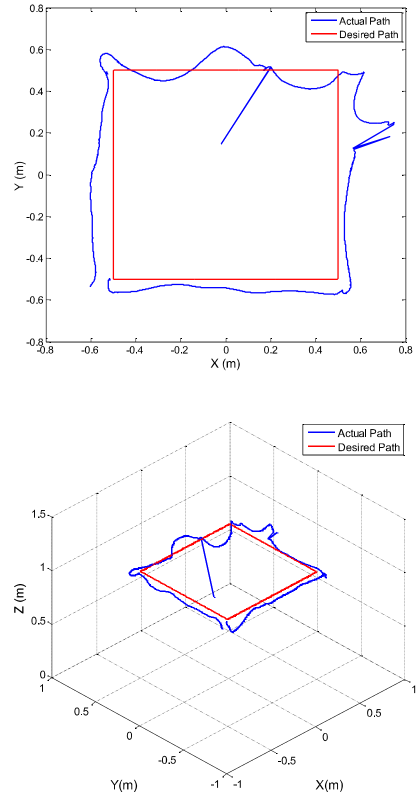

Figure 6. System response (LQR controller).

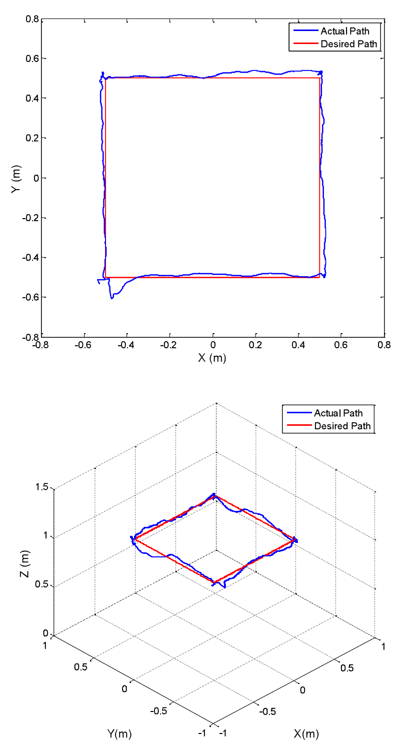

Figure 7. System response (LQR+MFC controller).

B. LQR+MFC Controller:

As shown above in Fig. 7, augmenting the MFC in the LQR controller revealed a significant improvement in control performance and trajectory tracking due to the capability of MFC in estimating the system uncertainties.

C. Degraded LQR Controller:

The objective of this experiment is to improve the performance of a badly tuned closed-loop system by adding the MFC. A bad selection of the Q and R matrices could lead to unwanted closed-loop system response. To degrade the closed-loop system performance, quantitatively, the gain matrix K is reduced by a given factor (35%), and this led to a poorer performance as shown in Fig. 8.

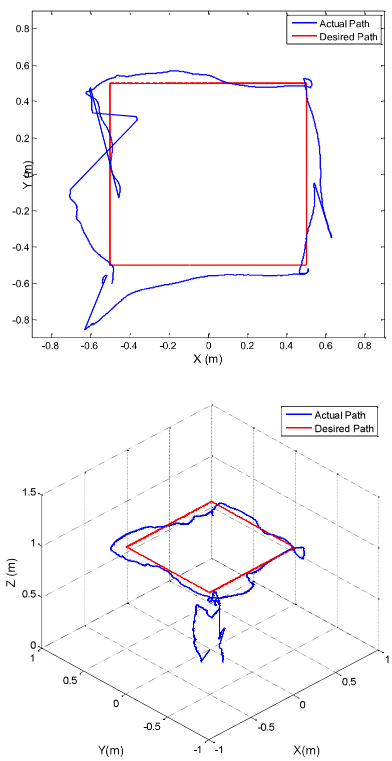

Figure 8. System response (degraded LQR controller).

D. Degraded LQR+MFC Controller:

The remedy of the degraded controller performance is done by adding the MFC. As shown in Fig. 9, the MFC compensate for the poor performance of the degraded controller and shows an excellent tracking performance.

A comparison between the different controller strategies is depicted in Fig. 10. The improvement of adding the MFC can be easily noticed in the 2D-drawing of Fig. 10.

Despite the fact that the quadrotor system is a highly MIMO nonlinear system, the performance of the controlled system using the MFC is outstanding, this is due to the powerful nature of MFC in compensating the system uncertainties and disturbances. The structured way of implementing the MFC in this paper has a crucial effect on the system performance.

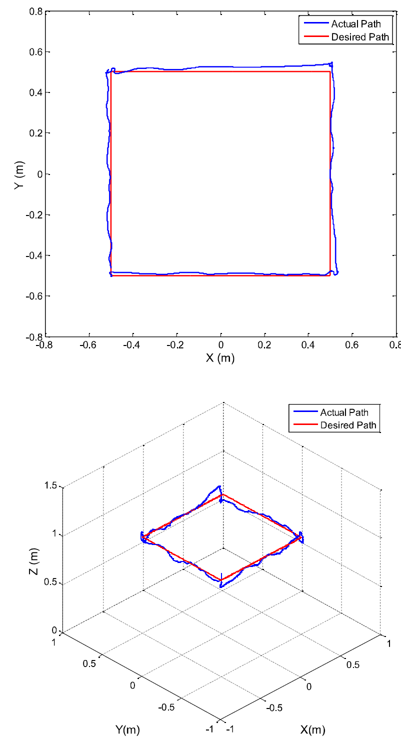

Figure 9. System response (degraded LQR+MFC controller).

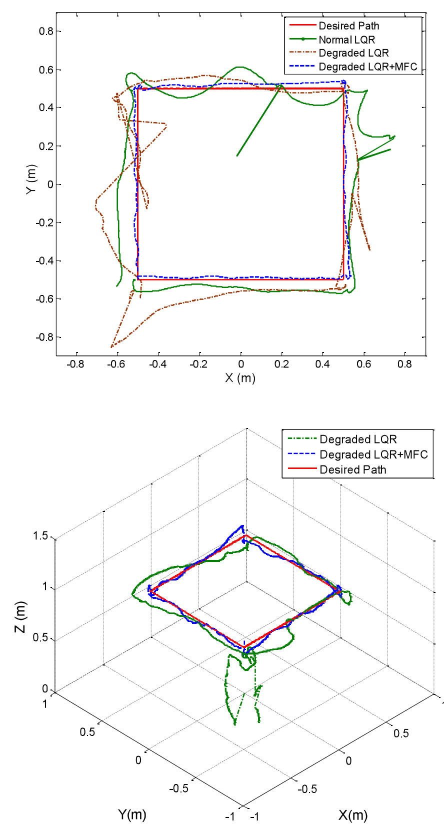

Figure 10. Comparsion between the different control techniques.

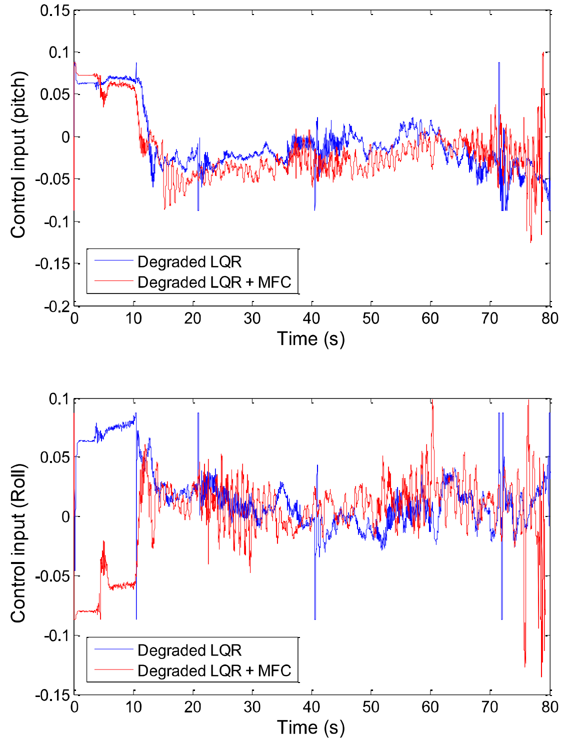

Figure 11. Pitch and roll control inputs of the degraded LQR controller vs degraded LQR+MFC controller.

Finally, as shown in Fig. 11, the control inputs generated from the planer position controllers (x, y), in the degraded LQR controller, are insufficient to keep the quadrotor on the desired trajectory (signals are running in a narrower control input range compared to the degraded LQR+MFC controller). In contrary, the MFC estimation to the system degradation and uncertainties in the degraded LQR+MFC controller shows faster and stronger efforts of the control inputs and therefore better tracking performance.

V. Conclusion

This paper uses the MFC technique and implementing it in a unique way to compensate for the un-modeled dynamics and system uncertainties of the Qball-X4 quadrotor. The challenge of using the MFC alongside with a highly MIMO nonlinear system is tackled using a structured way of decomposing the system into multi-SISO subsystems, and implementing the MFC on each one of them. Several flight test results of different control methods have validated the effect of augmenting the MFC algorithm and shown the system performance in controlling the quadrotor.

References

[1] G. Hoffmann, D. G. Rajnarayan, S. L. Waslander, D. Dostal, J. S. Jang, and C. J. Tomlin, "The Stanford testbed of autonomous rotorcraft for multi agent control (STARMAC)," in *Digital Avionics Systems Conference, 2004. DASC 04. The 23rd*, 2004, pp. 12. E. 4-121-10 Vol. 2.

[2] S. Bouabdallah, "Design and control of quadrotors with application to autonomous flying," *Lausanne Polytechnic University,* 2007.

[3] S. G. Fowers, "Stabilization and control of a quad-rotor micro-UAV using vision sensors," Brigham Young University. Department of Electrical and Computer Engineering, 2008.

[4] C. F. Petersen, H. Hansen, S. Larsson, L. B. Theilgaard Madsen, and M. Rimestad, "Autonomous Hovering with a Quadrotor Helicopter," *Aalborg: Aalborg Universitet,* 2008.

[5] T. Zhang, W. Li, M. Achtelik, K. Kuhnlenz, and M. Buss, "Multi-sensory motion estimation and control of a mini-quadrotor in an air-ground multi-robot system," in *Robotics and Biomimetics (ROBIO), 2009 IEEE International Conference on*, 2009, pp. 45-50.

[6] Q.-L. Zhou, Y. Zhang, C. Rabbath, and D. Theilliol, "Design of feedback linearization control and reconfigurable control allocation with application to a quadrotor UAV," in *Control and Fault-Tolerant Systems (SysTol), 2010 Conference on*, 2010, pp. 371-376.

[7] H. Bouadi, M. Bouchoucha, and M. Tadjine, "Sliding Mode Control based on Backstepping Approach for an UAV Type-Quadrotor," *International Journal of Applied Mathematics & Computer Sciences,* vol. 4, 2008.

[8] J. Colorado, A. Barrientos, A. Martinez, B. Lafaverges, and J. Valente, "Mini-quadrotor attitude control based on Hybrid Backstepping & Frenet-Serret theory," in *Robotics and Automation (ICRA), 2010 IEEE International Conference on*, 2010, pp. 1617-1622.

[9] M. Fliess and C. Join, "Model-free control and intelligent PID controllers: towards a possible trivialization of nonlinear control?," *15th IFAC Symposium on System Identification (SYSID 2009),* 2009.

[10] P.-A. Gédouin, C. Join, E. Delaleau, J.-M. Bourgeot, S. A. Chirani, and S. Calloch, "A new control strategy for shape memory alloys actuators," in *European Symposium on Martensitic Transformations*, 2009, p. 07007.

[11] L. Michel, C. Join, M. Fliess, P. Sicard, and A. Chériti, "Model-free control of dc/dc converters," in *Control and Modeling for Power Electronics (COMPEL), 2010 IEEE 12th Workshop on*, 2010, pp. 1-8.

[12] J. De Miras, C. Join, M. Fliess, S. Riachy, and S. Bonnet, "Active magnetic bearing: A new step for model-free control," in *52nd IEEE Conference on Decision and Control*, 2013.

[13] R. Madonski and P. Herman, "Model-free control of a two-dimensional system based on uncertainty reconstruction and attenuation," in *Control and Fault-Tolerant Systems (SysTol), 2013 Conference on*, 2013, pp. 542-547.

[14] M. Fliess and C. Join, "Model-free control," *International Journal of Control,* vol. 86, pp. 2228-2252, 2013.

[15] M. Mboup, C. Join, and M. Fliess, "Numerical differentiation with annihilators in noisy environment," *Numerical Algorithms,* vol. 50, pp. 439-467, 2009.

[16] Quanser, "QUARC® Real-Time Control Software," 13 January 2013. [Online]. Available: http://www.quanser.com/products/QUARC. [Accessed 10 December 2013].

[17] QUANSER, "User Manual," [Online]. Available: http://users.encs.concordia.ca/~realtime/coen421/doc/Quanser%20QBall-X4%20-%20User%20Manual.pdf. [Accessed 15 November 2013].

[18] I. D. Cowling, O. A. Yakimenko, J. F. Whidborne, and A. K. Cooke, "Direct method based control system for an autonomous quadrotor," *Journal of Intelligent & Robotic Systems,* vol. 60, pp. 285-316, 2010.