Fault-Tolerant Fuzzy Gain-Scheduled PID for a Quadrotor Helicopter Testbed in the Presence of Actuator Faults

Mohammad Hadi Amoozgar, Abbas Chamseddine, Youmin Zhang*

Concordia University, Montreal, Quebec, H3G 1M8, Canada
(e-mail: m_amoozg,abbasc,ymzhang@encs.concordia.ca)

Abstract: In the current study, an adaptive PID controller is proposed for fault-tolerant control of a quadrotor helicopter system in the presence of actuator faults. A fuzzy inference scheme is used to tune in real-time the controller gains. Tracking errors and change in tracking errors are used in this fuzzy scheduler to make the system act faster and more effectively in the event of fault occurrence. Two fault scenarios are investigated: the loss of control effectiveness in all actuators and the loss of control effectiveness in one single actuator. The proposed adaptive PID controller is compared with the conventional one through an experimental application to a quadrotor helicopter testbed at the Department of Mechanical and Industrial Engineering of Concordia University. The obtained results show the effectiveness of the proposed method.

Keywords: Autonomous aerial vehicles, Fuzzy control, PID control, Fault-tolerant systems, System failure and recovery.

1. INTRODUCTION

The quadrotor helicopter is relatively a simple, affordable and easy to fly system. It has been widely used to develop, implement and flight-test methods in control, fault diagnosis, fault tolerant control as well as multi-agent based technologies in formation flight, cooperative control, distributed control, surveillance and search missions, mobile wireless networks and communications. Some theoretical works consider the problems of control (Dierks and Jagannathan, 2008), formation flight (Dierks and Jagannathan, 2009) and fault diagnosis (Rafaralahy et al., 2008) of the quadrotor Unmanned Aerial Vehicle (UAV). However, few research laboratories are carrying out advanced theoretical and experimental works on the system. Among others, one may cite for example, the UAV health management project of the Aerospace Controls Lab. at MIT (SWARM, 2011), the Stanford Testbed of Autonomous Rotorcraft for Multi-Agent Control project (STARMAC, 2011) and the Micro Autonomous Systems Technologies project (MAST, 2011). A team of researchers is also currently working at the Department of Mechanical and Industrial Engineering of Concordia University to develop, implement and test approaches in Fault Detection and Diagnosis (FDD), Fault-Tolerant Control (FTC) and cooperative control with experimental application to the quadrotor unmanned helicopter system. For more information on the research activities carried out, interested readers are referred to the Networked Autonomous Vehicles (NAV) laboratory (NAV, 2011).

One of our main objectives is to propose control methods that are effective, simple to implement for real-time applications and robust to model uncertainties and external disturbances including actuator faults. PID (Proportional - Integral - Derivative) controllers are the most well-known controller in the society of automation and control, due to their simple structure and wide variety of usage. These kinds of controllers are classified into two main categories in terms of parameters selection strategies. In the first group, controller gains are fixed during operation while in the second group, gains change based on the operating conditions. In the first group, gains are tuned by the designer and remain invariable during the operation. One of the most well-known methods for choosing control gains in this group is Ziegler-Nichols method which has been addressed in lots of research works (Ziegler and Nichols, 1942). Although this method is simple and straightforward, fine tuning is required for different applications.

In most applications, due to structural changes the controlled system may lose its effectiveness, therefore the PID gains need to be continuously retuned during the system life span. To reduce the effort of retuning the gains and also in order to increase system's performance, in the second group of controllers, the gains are adapted online. Several methods have been proposed in the literature for PID gain scheduling. In Ng et al. (1997) a stable gain-scheduling PID controller is developed based on grid point concept for nonlinear systems, in which gains switch between some predefined values. Different gain scheduling methods were studied and compared in Karray et al. (2002). In Zhao et al. (1993) a new PID scheme is proposed in which the controller gains were scheduled by a fuzzy inference scheme. Many variations and improvements of this simple and effective method were followed by latter research works (Yu and Hsu, 2007; Zulfatman and Rahmat, 2009; Guo and Yang, 2010). A particle swarm optimization

* Corresponding Author.

method is used in Yu and Hsu (2007) to design membership functions of fuzzy PID controller. In Yao and Lin (2005), an accumulated genetic algorithm is proposed which learns the parameters and number of fuzzy rules in the fuzzy PID controller. An adaptive fuzzy PID using neural wavelet network is presented in El Emary et al. (2009). The interested readers can find a brief review of different fuzzy PID structures in Hu et al. (2001).

As part of our research group activities, a Gain-Scheduled PID (GS-PID) is designed for the quadrotor system in Sadeghzadeh et al. (2011). The GS-PID has been implemented for different sections of the entire flight envelope by properly tuning the PID controller gains for both normal and fault conditions. The switching from one PID to another is then based on the actuator's health status. It is worthy to note that the above method requires a Fault Detection and Diagnosis (FDD) scheme to provide the time of fault occurrence as well as the location and the magnitude of the fault during the flight. Motivated by this work and to eliminate the need for the FDD module, an adaptive PID controller is proposed in this paper for fault-tolerant control of a quadrotor helicopter system. A fuzzy inference scheme is used to tune in real-time the controller gains, where the tracking error and the change in tracking error are used in this fuzzy scheduler to make the system act faster and more effectively in the fault-free case as well as in the event of fault occurrence. Two fault scenarios are investigated: the loss of control effectiveness in all actuators and the loss of control effectiveness in one single actuator. The proposed PID controller is compared with the conventional one through an experimental application to the quadrotor helicopter testbed at the NAV Lab.

The reminder of this paper is as follows. Section 2 gives a description and the mathematical model of the quadrotor UAV testbed. Section 3 discusses the proposed fuzzy gain-scheduled PID controller. Some experimental results are illustrated in Section 4 before giving the concluding remarks.

2. DESCRIPTION AND DYNAMICS OF THE QUADROTOR UAV SYSTEM

The quadrotor UAV available at the NAV Lab is the Qball-X4 testbed (Figure 1) which was developed by Quanser Inc. partially under the financial support of NSERC (Natural Sciences and Engineering Research Council of Canada) in association with an NSERC Strategic Project Grant led by Concordia University since 2007.

The quadrotor UAV is enclosed within a protective carbon fiber round cage (therefore a name of Qball-X4) to ensure safe operation. It uses four 10-inch propellers and standard RC motors and speed controllers. It is equipped with the Quanser Embedded Control Module (QECM), which is comprised of a Quanser HiQ aero data acquisition card and a QuaRC-powered Gumstix embedded computer. The Quanser HiQ provides high-resolution accelerometer, gyroscope, and magnetometer IMU sensors as well as servo outputs to drive four motors. The on-board Gumstix computer runs QuaRC (Quanser's real-time control software), which allows to rapidly develop and deploy controllers designed in MATLAB/Simulink environment to control the Qball-X4 in real-time. The controllers run on-board

Fig. 1. The Quanser Qball-X4 quadrotor UAV

the vehicle itself and runtime sensors measurement, data logging and parameter tuning are supported between the host PC and the target vehicle.

The entire UAV system's block diagram is illustrated in Figure 2. It is composed of three main parts. The first part represents the Electronic Speed Controllers (ESCs) + the motors + the propellers in a set of four. The input to this part is $u = [u_1 \quad u_2 \quad u_3 \quad u_4]^T$ which are Pulse Width Modulation (PWM) signals. The output is the thrust vector $T = [T_1 \quad T_2 \quad T_3 \quad T_4]^T$ generated by four individually-controlled motor-driven propellers. The second part is the geometry that relates the generated thrusts to the applied lift and torques to the system. This geometry corresponds to the position and orientation of the propellers with respect to the center of mass of the Qball-X4. The third part is the dynamics that relate the applied lift and torques to the position (P), velocity (V) and acceleration (A) of the Qball-X4.

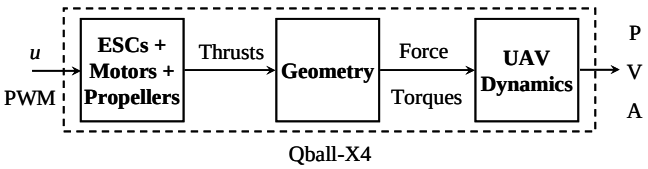

Fig. 2. The UAV system block diagram

The subsequent sections describe the corresponding mathematical model for each of the blocks of Figure 2.

2.1 ESCs, Motors and Propellers

The motors of the Qball-X4 are outrunner brushless motors. The generated thrust T_i of the i^{th} motor is related to the i^{th} PWM input u_i by a first-order linear transfer function:

$$T_i = K\frac{\omega}{s+\omega}u_i \quad ; \quad i = 1, ..., 4 \tag{1}$$

where K is a positive gain and ω is the motor bandwidth. K and ω are theoretically the same for the four motors but this may not be the case in practice. It should be noted that $u_i = 0$ correponds to zero thrust and that $u_i = 0.05$ corresponds to the maximal thrust that can be generated by the i^{th} motor.

2.2 Geometry

A schematic representation of the Qball-X4 is given in Figure 3. The motors and propellers are configured in such a way that the back and front (1 and 2) motors spin clockwise and the left and right (3 and 4) spin counter-clockwise. Each motor is located at a distance L from the center of mass o and when spinning, a motor produces a torque τ_i which is in the opposite direction of that of the motor as shown in Figure 3. The origin of the body-fixed frame is the system's center of mass o with the x-axis pointing from back to front and the y-axis pointing from right to left. The thrust T_i generated by the i^{th} propeller is always pointing upward in the z-direction in parallel to the motor's rotation axis. The thrusts T_i and the torques τ_i result in a lift in the z-direction (body-fixed frame) and torques about the x, y and z axis.

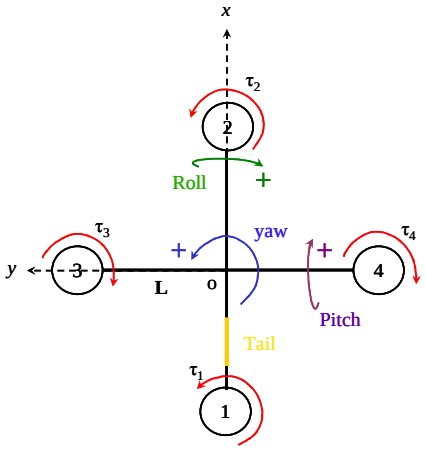

Fig. 3. Schematic representation of the Qball-X4

The relation between the lift/torques and the thrusts is:

$$\begin{aligned} u_z &= T_1 + T_2 + T_3 + T_4 \\ u_\theta &= L(T_1 - T_2) \\ u_\phi &= L(T_3 - T_4) \\ u_\psi &= \tau_1 + \tau_2 - \tau_3 - \tau_4 \end{aligned} \tag{2}$$

The torque τ_i produced by the i^{th} motor is directly related to the thrust T_i via the relation of $\tau_i = K_\psi T_i$ with K_ψ as a constant. In addition, by setting $T_i \approx K u_i$ from (1), the relation (2) can be written as:

$$\begin{bmatrix} u_z \\ u_\theta \\ u_\phi \\ u_\psi \end{bmatrix} = \begin{bmatrix} K & K & K & K \\ KL & -KL & 0 & 0 \\ 0 & 0 & KL & -KL \\ KK_\psi & KK_\psi & -KK_\psi & -KK_\psi \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix} \tag{3}$$

where u_z is the total lift generated by the four propellers and applied to the quadrotor UAV in the z-direction (body-fixed frame). u_θ, u_ϕ and u_ψ are respectively the applied torques in θ, ϕ and ψ directions which are the pitch, roll and yaw Euler angles respectively (see Figure 3). L is the distance from the center of mass to each motor.

2.3 UAV Dynamics

A commonly employed quadrotor UAV model (Xu and Ozguner, 2006) is:

$$\begin{aligned} m\ddot{x} &= u_z\,(cos\phi\ sin\theta\ cos\psi + sin\phi\ sin\psi)\,; & J_1\ddot{\theta} &= u_\theta \\ m\ddot{y} &= u_z\,(cos\phi\ sin\theta\ sin\psi - sin\phi\ cos\psi)\,; & J_2\ddot{\phi} &= u_\phi \\ m\ddot{z} &= u_z\,(cos\phi\ cos\theta) - mg; & J_3\ddot{\psi} &= u_\psi \end{aligned} \tag{4}$$

where x, y and z are the coordinates of the quadrotor UAV center of mass in the earth-frame. m is the mass and J_i $(i = 1, 2, 3)$ are the moments of inertia along y, x and z directions respectively.

A simplified linear model can be obtained by assuming hovering conditions ($u_z \approx mg$ in the x and y directions) with no yawing ($\psi = 0$) and small roll and pitch angles. This simplified model is given by:

$$\begin{aligned} \ddot{x} &= \theta g; & J_1\ddot{\theta} &= u_\theta \\ \ddot{y} &= -\phi g; & J_2\ddot{\phi} &= u_\phi \\ \ddot{z} &= u_z/m - g; & J_3\ddot{\psi} &= u_\psi \end{aligned} \tag{5}$$

Although that a PID controller does not require a mathematical model of the controlled process, this simplified model given in (5) is employed to calculate an optimal set of PID gains using LQR techniques. The performance of the obtained PID controller is compared to that of the fuzzy PID presented in the subsequent section.

3. FUZZY GAIN-SCHEDULED PID CONTROLLER

The transfer function of a conventional PID controller is:

$$G(s) = K_p + \frac{K_i}{s} + K_d s \tag{6}$$

where K_p, K_i, and K_d are the proportional, integral, and derivative gains, respectively. Conventional PID controllers are frequently and widely used in vast number of industrial applications. They are simple and easy to use due to the fact that they do not need any mathematical model of the controlled process or complicated theories. But one of the main drawbacks of these controllers is that there is no certain way for choosing the control parameters which guarantees the good performance. Although PID controllers are robust against structural changes and uncertainties in the system parameters, their performance may be affected by such changes or may even lead to system instability. Therefore in real world applications these gains need to be fine-tuned to keep the required performance. To overcome this shortcoming, Fuzzy Logic Controller (FLC) is used to tune PID gains online where the tracking error and the change of the tracking error are used to determine control parameters.

Controller gains can be calculated through a simple linear transformation:

$$K_p = (K_{p,\max} - K_{p,\min})\,K'_p + K_{p,\min} \tag{7}$$

$$K_i = (K_{i,\max} - K_{i,\min})\,K'_i + K_{i,\min} \tag{8}$$

$$K_d = (K_{d,\max} - K_{d,\min})\,K'_d + K_{d,\min} \tag{9}$$

with $[K_{p,\min}, K_{p,\max}]$, $[K_{i,\min}, K_{i,\max}]$ and $[K_{d,\min}, K_{d,\max}]$ are predefined ranges of K_p, K_i, and K_d respectively. A set of linguistic rules in the form of (10) is used in the FLC structure to determine K'_p, K'_i and K'_d:

$$\text{If } e(k) \text{ is } A_i \text{ and } \Delta e(k) \text{ is } B_i \text{ then } K'_p \text{ is } C_i,\ K'_i \text{ is } D_i,\ \text{and } K'_d \text{ is } E_i \tag{10}$$

where A_i, B_i, C_i, D_i, and E_i are fuzzy sets corresponding to $e(k)$, $\Delta e(k)$, K'_p, K'_i, and K'_d respectively. Three sets of 49 rules are used to determine controller gains. Tables 1-3 show the linguist rules used in the FLC. In these tables, N, P, ZO, S, M, B represent negative, positive, approximately zero, small, medium, and big respectively. For example NB means Negative Big, and so on.

Table 1. Fuzzy tuning rules for K'_p

		$\Delta e(k)$						
		NB	**NM**	**NS**	**ZO**	**PS**	**PM**	**PB**
	NB	B	B	B	B	B	B	B
	NM	S	B	B	B	B	B	S
	NS	S	S	B	B	B	S	S
$e(k)$	**ZO**	S	S	S	B	S	S	S
	PS	S	S	B	B	B	S	S
	PM	S	B	B	B	B	B	S
	PB	B	B	B	B	B	B	B

Table 2. Fuzzy tuning rules for K'_i

		$\Delta e(k)$						
		NB	**NM**	**NS**	**ZO**	**PS**	**PM**	**PB**
	NB	S	S	S	S	S	S	S
	NM	B	B	S	S	S	B	B
	NS	B	B	B	S	B	B	B
$e(k)$	**ZO**	B	B	B	B	B	B	B
	PS	B	B	B	S	B	B	B
	PM	B	B	S	S	S	B	B
	PB	S	S	S	S	S	S	S

Table 3. Fuzzy tuning rules for K'_d

		$\Delta e(k)$						
		NB	**NM**	**NS**	**ZO**	**PS**	**PM**	**PB**
	NB	B	B	B	B	B	B	B
	NM	M	M	B	B	B	M	M
	NS	S	M	M	B	M	M	S
$e(k)$	**ZO**	ZO	S	M	B	M	S	ZO
	PS	S	M	M	B	M	M	S
	PM	M	M	B	B	B	M	M
	PB	B	B	B	B	B	B	B

The membership functions for input variables are defined with triangular and trapezoidal shapes and those for output variables are singleton (Figures 4 and 5). All the fuzzy sets for input and output values are normalized for convenience.

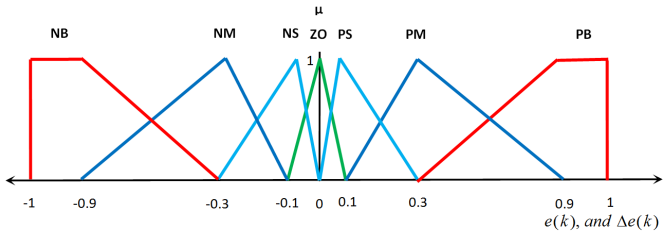

Fig. 4. Membership function for $e(k)$ and $\Delta e(k)$

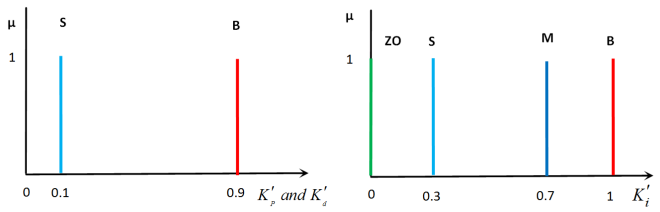

Fig. 5. Membership function for K'_p, K'_d and K'_i

The generated surfaces for the FLC are shown in Figures 6-8.

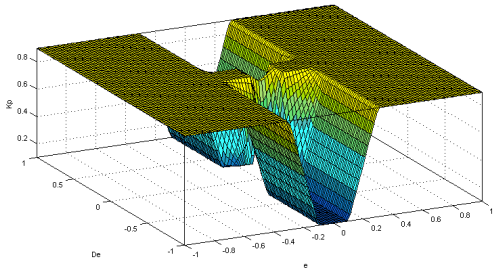

Fig. 6. Surface for K'_p

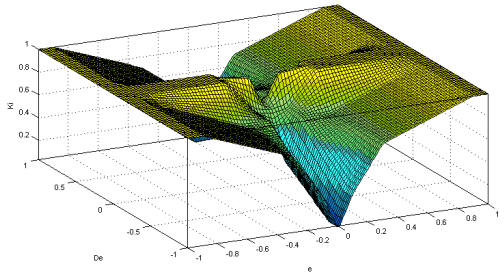

Fig. 7. Surface for K'_i

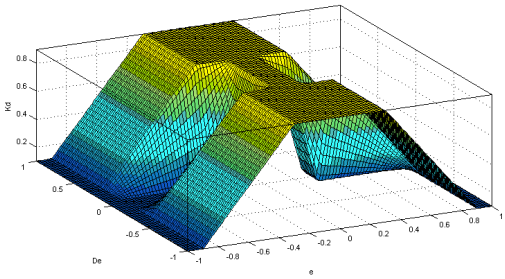

Fig. 8. Surface for K'_d

4. EXPERIMENTAL RESULTS

The fuzzy PID controller proposed in Section 3 has been experimentally tested on the Qball-X4 testbed. The controller is built using Matlab/Simulink and downloaded on the Gumstix emdedded computer to be run on-board with a frequency of 200 Hz. The experiments are taking place indoor in the absence of GPS signals and thus the OptiTrack camera system from NaturalPoint is employed to provide the system position in the 3D space. In all

experiments, the system is required to hover at an altitude of 1 m and the faults are taking place at time instant $t = 20$ s.

4.1 First Fault Scenario

In the first fault scenario, it is assumed that a loss of control effectiveness of 15% is taking place in the four motors. This kind of fault results in a loss of altitude and does not really produce significant movement along the x or y directions. The gains of the conventional PID for the height control are $K_p = 0.0122$, $K_i = 0.0079$, and $K_d = 0.0093$. The predefined ranges of K_p, K_i, and K_d for the fuzzy gain-scheduled PID in the height control are $K_{p,\min} = 0.010$, $K_{p,\max} = 0.015$, $K_{i,\min} = 0.007$, $K_{i,\max} = 0.010$, $K_{d,\min} = 0.0085$, and $K_{d,\max} = 0.0095$. Figure 9 shows a comparison between the conventional and the fuzzy adaptive PID controllers for the height holding flight. It is clear that the fuzzy adaptive PID controller reduces the fault effect on the system by reacting faster and returning the system quicker to its hovering position.

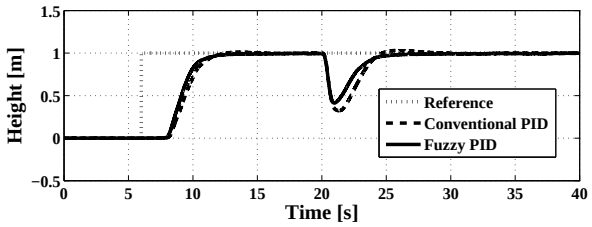

Fig. 9. Comparison between conventional and fuzzy PID

The time evolutions of the fuzzy PID gains are illustrated in Figure 10. Unlike those of the conventional PID, the fuzzy gains are time-varying to adapt to uncertainties, disturbances and faults as can be clearly seen at $t = 20$ s.

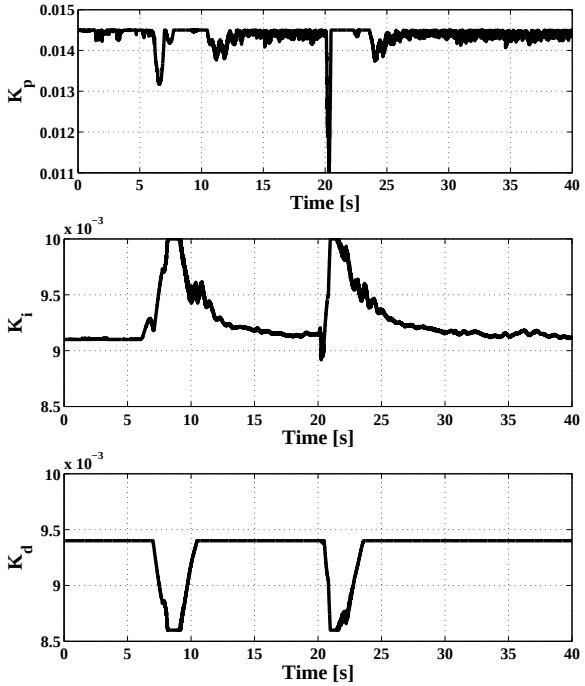

Fig. 10. Gains K_p, K_i, and K_d in the first scenario

It can be seen in Figure 10 that after the fault occurs, K_p decreases to avoid system overshoot due to increase in tracking error. The derivative gain K_d remains fixed with a high value to make a fast response to sudden changes in tracking error. When the system stops descending (loosing altitude) K_d decreases to let the system recovers faster and goes back to its desired position. After the fault, integrator gain K_i also increased to help the recovery process.

Table 4 gives a quantitative comparison between the conventional and the fuzzy PID. The Root Mean Square (RMS) is calculated for the tracking error before fault occurrence and for the 5 seconds after fault. One can see that before fault occurrence, the performance of both controllers are close. However, in the fault-case the fuzzy PID greatly reduces tracking error.

Table 4. RMS of tracking error

	Before Fault		**After Fault**	
	Conv.	**Fuzzy**	**Conv.**	**Fuzzy**
z-direction	88×10^{-4}	84×10^{-4}	127×10^{-4}	98×10^{-4}

4.2 Second Fault Scenario

In the second fault scenario, it is assumed that a loss of control effectiveness of 20% is taking place in the third motor. This kind of fault results in a loss of altitude and drift along the y direction. The gains and predefined ranges for the PID controllers along the z-direction remain the same as given in the previous section. The gains of the conventional PID for the y-direction are $K_p = 0.2137$, $K_i = 0.258$, and $K_d = 0.238$. The predefined ranges of K_p, K_i, and K_d for the fuzzy PID in the y-direction are $K_{p,\min} = 0.09$, $K_{p,\max} = 0.35$, $K_{i,\min} = 0.13$, $K_{i,\max} = 0.35$, $K_{d,\min} = 0.023$, and $K_{d,\max} = 0.029$. Figure 11 shows a comparison between the conventional and the fuzzy PID controllers along the height and y-direction. As in the first scenario, the fuzzy PID allows the system to react and return faster to its hovering position. The time evolutions of the fuzzy PID gains are illustrated in Figure 12. These gains are related to the fuzzy PID controller in the y-direction.

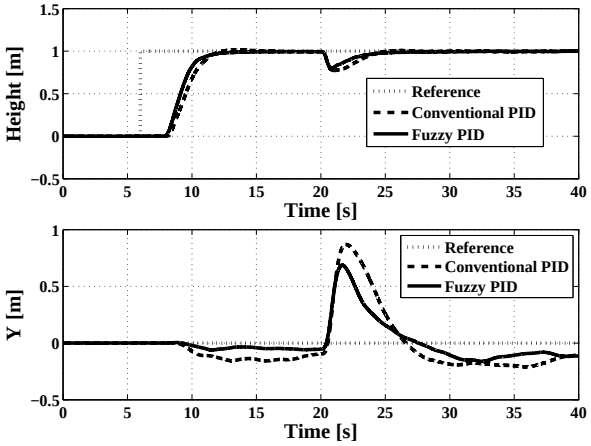

Fig. 11. Comparison between conventional and fuzzy PID

As in the first scenario, Table 5 gives an quantitative comparison between both controllers using the RMS of the

tracking errors along z and y directions. Once again, the fuzzy PID improves system's performance specially when faults occur.

Table 5. RMS of tracking errors

	Before Fault		After Fault	
	Conv.	**Fuzzy**	**Conv.**	**Fuzzy**
z-direct.	89×10^{-4}	84×10^{-4}	44×10^{-4}	32×10^{-4}
y-direct.	21×10^{-4}	7.6×10^{-4}	191×10^{-4}	132×10^{-4}

5. CONCLUSION

This paper addressed the design of fuzzy gain-scheduled PID controller for a quadrotor unmanned helicopter in the presence of actuator faults. The proposed controller has been experimentally tested and compared with the conventional one. The obtained results revealed the effectiveness of the proposed method and its ability to adapt in the presence of uncertainties and external disturbances.

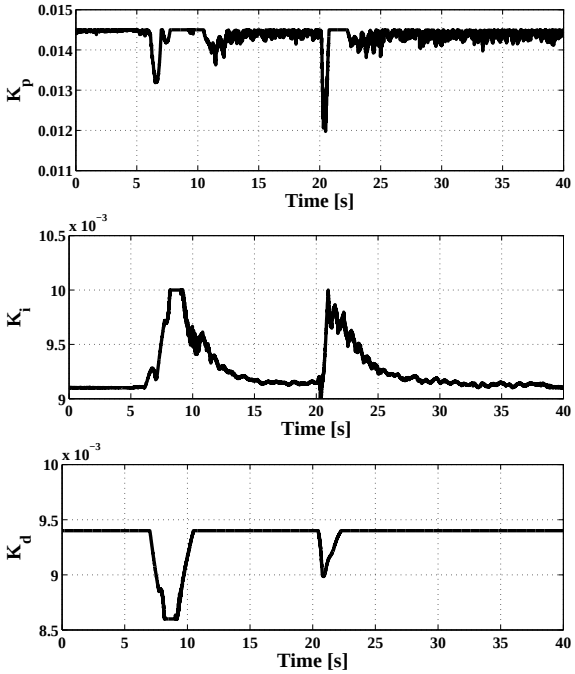

Fig. 12. Gains K_p, K_i, and K_d in the second scenario

ACKNOWLEDGEMENTS

This work is supported by the Natural Sciences and Engineering Research Council of Canada (NSERC) Postdoctoral Fellowship (PDF) program to the second author and the NSERC Strategic Project Grant and Discovery Project Grant led by the third author. Support from Quanser Inc. and colleagues from Quanser Inc. for the development of the Qball-X4 UAV test-bed is also highly appreciated.

REFERENCES

Dierks, T. and Jagannathan, S. (2008). Neural network output feedback control of a quadrotor UAV. In *Proceedings of the 47th IEEE Conference on Decision and Control*, 3633–3639. Cancun, Mexico.

Dierks, T. and Jagannathan, S. (2009). Neural network control of quadrotor UAV formations. In *American Control Conf.*, 2990–2996. St. Louis, Missouri, USA.

El Emary, I.M.M., Emar, W., and Aqel, M.J. (2009). The adaptive fuzzy designed PID controller using wavelet network. *Journal of Computer Science and Information System*, 6(2), 141–163.

Guo, Y. and Yang, T. (2010). A new type of computational verb gain-scheduling PID controller. In *International Conference on Counterfeiting Security and Identification in Communication*, 235–238. Chengdu.

Hu, B., Mann, G.K.I., and Gosine, R.G. (2001). A systematic study of fuzzy PID controllers-function-based evaluation approach. *IEEE Transactions on Fuzzy Systems*, 9(5), 699–712.

Karray, F., Gueaieb, W., and Al-Sharhan, S. (2002). The hierarchical expert tuning of PID controllers using tools of soft computing. *IEEE Transactions on Systems, Man, and Cybernetics-Part B: Cybernetics*, 32(1), 77–90.

MAST (2011). `https://www.grasp.upenn.edu/research/micro_autonomous_system_technologies_mast`. [cited 30 October 2011].

NAV (2011). `http://users.encs.concordia.ca/~ymzhang/UAVs.htm`. [cited 30 October 2011].

Ng, T.C.T., Leung, F.H.F., and Tam, P.K.S. (1997). A simple gain scheduled PID controller with stability consideration based on a grid-point concept. In *Proceedings of the IEEE International Symposium on Industrial Electronics*, 1090–1094. Guimaraes, Portugal.

Rafaralahy, H., Richard, E., Boutayeb, M., and Zasadzinski, M. (2008). Simultaneous observer based sensor diagnosis and speed estimation of unmanned aerial vehicle. In *Proceedings of the 47th IEEE Conference on Decision and Control*, 2938–2943. Cancun, Mexico.

Sadeghzadeh, I., Mehta, A., Zhang, Y.M., and Rabbath, C.A. (2011). Fault-tolerant trajectory tracking control of a quadrotor helicopter using gain-scheduled PID and model reference adaptive control. In *Annual Conference of the PHM Society*. Montreal, QC, Canada.

STARMAC (2011). `http://hybrid.stanford.edu/~starmac/project.htm`. [cited 30 October 2011].

SWARM (2011). `http://vertol.mit.edu`. [cited 30 October 2011].

Xu, R. and Ozguner, U. (2006). Sliding mode control of a quadrotor helicopter. In *Proceedings of the 45th IEEE Conference on Decision and Control*, 4957–4962. San Diego, California, USA.

Yao, L. and Lin, C. (2005). Design of gain scheduled fuzzy PID controller. *World Academy of Science, Engineering and Technology*, (1), 152–156.

Yu, K. and Hsu, J. (2007). Fuzzy gain scheduling PID control design based on particle swarm optimization method. In *Second International Conference on Innovative Computing, Information and Control*. Kumamoto.

Zhao, Z., Tomizuka, M., and Isaka, S. (1993). Fuzzy gain scheduling of PID controllers. *IEEE Transactions on Systems, Man, and Cybernetics*, 23(5), 1392–1398.

Ziegler, J.G. and Nichols, N.B. (1942). Optimum settings for automatic controllers. *ASME Trans.*, (64), 759–768.

Zulfatman and Rahmat, M.F. (2009). Application of self-tuning fuzzy PID controller on industrial hydraulic actuator using system identification approach. *Int. J. on Smart Sensing and Intelligent Systems*, 2, 246–261.