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

A Learning-Based Fuzzy LQR Control Scheme for Height Control of An Unmanned Quadrotor Helicopter*

Z. X. Liu[1], C. Yuan[2], Y. M. Zhang[3] and J. Luo[4]

Abstract— **In this paper, a novel learning-based fuzzy Linear Quadratic Regulator (LQR) control method using Extended Kalman Filter (EKF) to optimize a Mamdani fuzzy LQR controller is presented. The EKF is used to adjust the shape of membership functions and rules of the fuzzy controller to adapt with the working conditions automatically during the operation process to minimize the control error. Then, the LQR controller is tuned according to the modified fuzzy membership functions and rules. The proposed approach in this paper is verified by testing and comparing performance of the height control problem of an unmanned quadrotor helicopter between the conventional LQR and learning-based fuzzy LQR controllers in the Matlab/Simulink. Simulation results show that developed method is effective for online optimization of fuzzy LQR controllers, improving control performance significantly.**

I. INTRODUCTION

As a relatively affordable, simple and easy to fly platform, the quadrotor helicopter has been widely employed to develop and implement flight-test approaches in control, fault diagnosis, fault tolerant control [13], multi-agent based techniques in cooperative [21], formation and distributed control [4], surveillance and search missions [23] as well as mobile wireless networks and communications [22]. However, few research laboratories are carrying out advanced theoretical and experimental works on the quadrotor Unmanned Aerial Vehicle (UAV) system [14].

In modern control theory, LQR is a widely used approach that analyzes a system by employing state-space method. Using state space method, it is relatively easier to work with a multi-input and multi-output system. The system can be stabilized using full-state feedback system by designing a LQR controller which can be developed to determine the value of the gain of the state feedback control [4]. Several existing LQR methods have been addressed in the literature for quadrotor helicopter design. In [15], a LQR controller for the vertical flight mode has been developed and its performance has been tested with several simulations. In [16], the authors developed a LQR controller and applied it to the vertical flight mode for all possible yaw angles which has been shown to be quite effective. The optimal guidance law of the quadrotor helicopter, using LQR is presented in [17] which is based on linearizing the dynamic equations of the quadrotor in different operation conditions. A nonlinear dynamic model based on quaternions for attitude was proposed and its corresponding LQR gain scheduling control were given in [18] where a good performance of trajectory tracking and attitude control of a quadrotor was achieved. An application of LQR in obstacle avoidance that defines sets of control inputs which leads a robot to avoid collision with obstacles was reported in [19], then this method was extended for reciprocal collision avoidance among multiple robots . It is worth mentioning that in [20], the authors employed LQR combined with Kalman filter to achieve motion control in pitch, roll, and yaw directions.

Motivated by the works in [3] and [4], this paper designed a learning-based fuzzy LQR controller for tuning the state feedback gains of the quadrotor helicopter on-line. This controller is used for height control of an unmanned quadrotor helicopter by step response and partial loss of control effectiveness respectively. The controller simultaneously makes use of learning function of EKF, the effectiveness of fuzzy logic control and the good performance of LQR.

One of contributions of this presented paper is using EKF as a learning method to tune the membership functions and rules of Mamdani fuzzy LQR on-line. In addition, this paper simulated this control scheme on the height control of an unmanned quadrotor helicopter, compared its performance with a conventional LQR method and verified its improved performance.

This paper is organized as follows: The control system structure is stated in the next section. In Section 3, the dynamical equations of the unmanned quadrotor helicopter which is used for testing the proposed control approach are presented. Simulation results for illustrating the effectiveness of the proposed control method are given in Section 4. Some conclusion remarks are drawn in the last section.

II. CONTROL SYSTEM STRUCTURE

A. Linear Quadratic Regulator

The well-known LQR approach might be a suitable option [6]. Assume a system can be described in the following state space form:

$$\dot{x}(t) = Ax(t) + Bu(t) \tag{1}$$

where $x(t) \in \Re^n, u(t) \in \Re^m, A \in \Re^{n\times n}, B \in \Re^{n\times m}$. The objective of LQR is to find an appropriate control input $u(t)$ to force the system from any initial state $x(t_0)$ to the equivalent state $\lim_{t\to\infty} x(t) = 0$ within an infinite time interval

[1]Z. X. Liu is the PhD student with Faculty of Mechanical Engineering, University of Concordia, Canada, l_zhixia@encs.concordia.ca
[2]C. Yuan is the PhD student with Faculty of Mechanical Engineering, University of Concordia, Canada, chi_yua@encs.concordia.ca
[3]Y. M. Zhang is the Associate Professor with Faculty of Mechanical Engineering, University of Concordia, Canada, ymzhang@encs.concordia.ca
[4]J. Luo is the Professor with School of Mechatronic Engineering and Automation, Shanghai University, China, luojun@shu.edu.cn

by minimizing a quadratic equation of the form:

$$J = \int_{t_0}^{\infty} (x(t)^T Qx(t) + u(t)^T Ru(t))dt \tag{2}$$

where $Q \in \Re^{n\times n}$ is a symmetric matrix, at least positive semidefinite and $R \in \Re^{n\times n}$ is a symmetric positive definite matrix as well. Then the state feedback gain K can be obtained by solving algebraic Riccati equations [5]. And the optimal control input can be computed as follows:

$$u(t) = -Kx(t) \tag{3}$$

B. Fuzzy System

Mamdani Fuzzy Logic Control (FLC) is conceived as an excellent method for many control system applications with non-linearity, real-time and complex mathematical computation need [2].

In this paper, error "$e(t)$" and derivative of error "$\dot{e}(t)$" are chosen to be two inputs to the fuzzy controllers. For each inputs, the triangle membership functions are used and its mathematical expression is described as follows:

$$f_{ij}(z_j) = \begin{cases} 1 + (z_j - c_{ij})/b_{ij}^- & if \ -b_{ij}^- \leq (z_j - c_{ij}) \leq 0 \\ 1 - (z_j - c_{ij})/b_{ij}^+ & if \ 0 \leq (z_j - c_{ij}) \leq b_{ij}^+ \\ 0 & otherwise, \ j = 1, 2, ..., N \end{cases} \tag{4}$$

where z_j is the jth input, c_{ij}, b_{ij}^- and b_{ij}^+ are the centroid, lower half-width, and upper half-width of the ith triangle membership function of the jth input, respectively. N is the number of triangles.

Max-min aggregation method and Centroid defuzzification as one of the popular defuzzification methods are used to compute the control output with a pair of inputs. Furthermore, it is assumed that there is only one output in this fuzzy system. Similar to input, the defuzzification rule for output can be given as:

$$m_j(y) = \begin{cases} 1 + (y - \gamma_j)/\beta_j^- & if \ -\beta_j^- \leq (y - \gamma_j) \leq 0 \\ 1 - (y - \gamma_j)/\beta_j^+ & if \ 0 \leq (y - \gamma_j) \leq \beta_j^+ \\ 0 & otherwise. \end{cases} \tag{5}$$

where $m_j(y)$ is the jth fuzzy output, γ_j, y, β_j^- and β_j^+ are the modal point, crisp number, lower half-width, and upper half-width of the jth output rule respectively. The jth rule is a result of z_1 which is based on fuzzy set i and z_2 is based on fuzzy set k. Then the activation grade of the result of the jth rule can be denoted by w_j, which can be illustrated as:

$$w_j = min[f_{i1}(z_1), f_{k2}(z_2)] \tag{6}$$

So the fuzzy output while $z_1 \in$ fuzzy set i and $z_2 \in$ fuzzy set k is addressed as:

$$\bar{m}_j(y) = w_j m_j(y) \tag{7}$$

The whole fuzzy output $m(y)$ is proposed as:

$$m(y) = \sum_{j=1}^{M} \bar{m}_j(y) \tag{8}$$

By using centroid defuzzification technique, the fuzzy output can be mapped to a crisp number $\hat{y}$ as below:

$$\hat{y} = \frac{\sum_{j=1}^{M} \omega_j C_j S_j}{\sum_{j=1}^{M} \omega_j S_j} \tag{9}$$

where C_j and S_j are the centroid and area of the jth fuzzy membership function of output. The centroid C_j is defined as:

$$C_j = \frac{\int y m_j(y)dy}{\int m_j(y)dy} \tag{10}$$

Since the fuzzy membership functions here are assumed to be triangle, thus the error function is can be given by:

$$E = \frac{1}{2N} \sum_{n=1}^{N} g_n E_n^2 E_n = \hat{y}_n - y_n \tag{11}$$

where g_n, $\hat{y}_n$, y_n and N are time-dependent weighting function, the actual output of the fuzzy system, the target output of the fuzzy system and the amount of training samples respectively.

C. Extended Kalman Filter (EKF)

Kalman filter is a powerful mathematical tool for stochastic estimation from noisy sensor measurements. A Kalman filter that linearizes about the current mean and covariance is referred to as an extended Kalman filter which has been widely applied in many engineering fields and control system designs [12].

In this paper, EKF is used for on-line optimization of the fuzzy membership functions. Assume that the stochastic variable x_i is the state of a system at time $t_i (i = 1, 2, ...)$. In next stage, the state can be denoted by a stochastic difference equation [9]:

$$\begin{aligned} x_{i+1} &= f(x_i) + \omega_i \\ \omega_i &\sim N(0, Q_i) \end{aligned} \tag{12}$$

and the measurement vector y_i which associates with the system state is expressed as:

$$\begin{aligned} y_i &= h(x_i) + \nu_i \\ \nu_i &\sim N(0, R_i) \end{aligned} \tag{13}$$

where ω_i, ν_i, Q_i and R_i are the process noise, measurement noise, process noise covariance of ω_i and the measurement noise covariance of ν_i respectively. $f(\cdot)$ and $h(\cdot)$ represent nonlinear vector functions of the state. The desired estimate $\hat{x}_i$ can be calculated by the EKF recursive equations:

$$\begin{aligned} F_i &= \frac{\partial f(x)}{\partial x}\Big|_{x=\hat{x}_i} \\ H_i &= \frac{\partial h(x)}{\partial x}\Big|_{x=\hat{x}_i} \\ K_i &= P_i H_i^T (R_i + H_i P_i H_i^T)^{-1} \\ \hat{x}_i &= f(\hat{x}_{i-1}) + K_i[d_{i-1} - h(\hat{x}_{i-1})] \\ P_{i+1} &= F_i(P_i - K_i H_i P_i)F_i^T + Q_i \end{aligned} \tag{14}$$

where d_i is the observation vector, K_i is the well-known Kalman gain. P_i is the covariance matrix of the state estimation error, and the estimated state $\hat{x}_{i+1}$ is the optimal solution that approaches the conditional mean value $E[x_{i+1}|(d_0, d_1, \ldots, d_i)]$.

D. Learning-Based Fuzzy LQR

The traditional LQR controllers cannot achieve reasonable performance over a wide region of operating conditions since the feedback state gains are fixed.

Motivated by the successful use of fuzzy LQR [8] as well as EKF for training model predictive control [7] and fuzzy logic control [3], a similar approach is applied to train the fuzzy LQR systems to determine best values for parameters in fuzzy control rules in which the control performance can be improved.

In this research, the system is considered as a single input and single output control problem. The main idea is to design a supervisory fuzzy controller capable of adjusting the controller parameters in order to obtain the desired response. The reason behind this scheme is to combine the best features of EKF, fuzzy control and those of the optimal LQR. On top of that, to synthesize a fuzzy controller, we pursue the idea of making it suitable to the LQR with small inputs since the LQR was so successful. At the same time, we still have the additional tuning flexibility with the smart fuzzy controller to reshape its control surface so that it can still perform differently from the conventional LQR even for the larger inputs.

The optimization of fuzzy membership functions can be viewed as a weighted least-squares minimization problem. In this research, a two-input and one-output fuzzy system is taken into account. Assume that this fuzzy system has n fuzzy sets for the first input, m fuzzy sets for the second input, and k fuzzy sets for the output. As Eqs. (4) and (5) denoted, the state of the nonlinear system can be represented like:

$$\begin{aligned} x = [&b_{11}^- \; b_{11}^+ \; c_{11} \; \ldots \; b_{n1}^- \; b_{n1}^+ \; c_{n1} \\ &b_{12}^- \; b_{12}^+ \; c_{12} \; \ldots \; b_{m2}^- \; b_{m2}^+ \; c_{m2} \\ &\beta_1^- \; \beta_1^+ \; \gamma_1 \; \ldots \; \beta_k^- \; \beta_k^+ \; \gamma_k]^T \end{aligned} \tag{15}$$

The vector x as the state (x_i) of the EKF Eq. (12) which consists of all the fuzzy membership function parameters. The fuzzy system's nonlinear mapping from the membership function parameters to the fuzzy system's output is $h(x_i)$. Then by applying the Kalman recursive equation Eq. (14), assigning the output of the fuzzy system, the mapping, the current membership function parameters and the partial derivative of the fuzzy output to be d_i, $f(\cdot)$, $h(\hat{x}_i)$ and H_i, respectively. F_i is an identity matrix.

Based on the on-line tuned EKF fuzzy logic knowledge, the learning-based fuzzy LQR (as described in Fig. 1) tuner takes error "e" (the error between reference input and the system output) and derivation of the error "$\dot{e}$" as the inputs to the fuzzy controller. Then, in order to meet the different requirements of producing a reasonable performance, the fuzzy input and output rules (Eqs. (4) and (5)) can be made use of to establish the fuzzy tuning parameters and use the

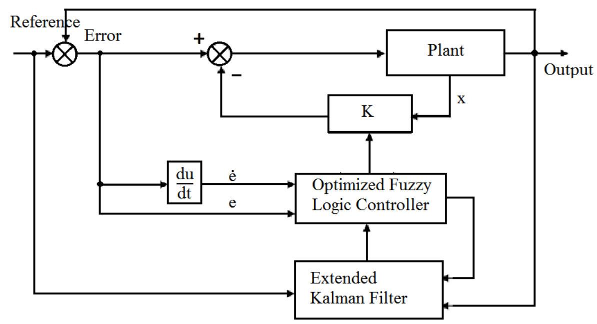

Fig. 1. Learning-based fuzzy LQR controller

following Eq. (16) to modify the feedback state gain K on-line

$$K = K_0 + T\Delta K, T \in [0, 1] \tag{16}$$

where T is the tuning parameter obtained from the output of the learning-based fuzzy controllers which is equal to the result of Eq. (9), $\Delta K = K_{max} - K_{min}$ is the allowable tuning range of K, K_0 is the value from conventional LQR, K_{max} and K_{min} are maximum and minimum tuning values of K respectively.

Consequently, the feedback state gain K is added into the feedback control system to get the optimal control input by Eq. (3).

III. DYNAMICAL EQUATIONS OF UNMANNED QUADROTOR HELICOPTER

A. Nonlinear Model of a Quadrotor Helicopter

According to the detailed description of the dynamical equations of a quadrotor helicopter in [10] and [11], the quadrotor helicopter with respect to the earth-fixed coordinate system can be represented as:

$$\begin{aligned} \ddot{x} &= \frac{(sin\psi sin\phi + cos\psi sin\theta cos\phi)u_1 - K_1\dot{x}}{m} \\ \ddot{y} &= \frac{(sin\psi sin\theta cos\phi - cos\psi sin\phi)u_1 - K_2\dot{y}}{m} \\ \ddot{z} &= \frac{(cos\theta cos\phi)u_1 - K_3\dot{z}}{m} - g \\ \ddot{\phi} &= \frac{u_3 l - K_4\dot{\phi}}{I_z} \\ \ddot{\theta} &= \frac{u_2 l - K_5\dot{\theta}}{I_y} \\ \ddot{\psi} &= \frac{u_4 c - K_6\dot{\psi}}{I_z} \end{aligned} \tag{17}$$

where $K_n (n = 1, 2, ..., 6)$, l, and c are the drag coefficients related with the aerodynamic drag force, the center distance between the gravity of the quadrotor helicopter and each propeller, and the thrust-to-moment scaling factor respectively. I_x, I_y, and I_z represent the moments of inertia along x, y, and z directions. The movement vector components u_1, u_2, u_3, u_4 are respectively the throttle, roll, pitch and yaw that is:

$$\begin{bmatrix} u_1 \\ u_2 \\ u_3 \\ u_4 \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 1 \\ 0 & -1 & 0 & 1 \\ -1 & 0 & 1 & 0 \\ 1 & -1 & 1 & -1 \end{bmatrix} \begin{bmatrix} F_1 \\ F_2 \\ F_3 \\ F_4 \end{bmatrix} \quad (18)$$

The thrust F_i $(i = 1, 2, 3, 4)$ of each rotor is generated by the pulse-width modulation (PWM) input. Their relationship is:

$$F_i = K_{motor} \frac{\omega_{motor}}{s + \omega_{motor}} PWM_i \quad (19)$$

where ω_{motor} denotes actuator bandwidth, K_{motor} is the gain and $PWM_i (i = 1, 2, 3, 4)$ represents PWM signals. Table I shows the nominal system parameters of the quadrotor helicopter used in the simulation.

TABLE I
THE SYSTEM PARAMETERS USED IN THE SIMULATION

Parameter	Value	Unit
ω_{motor}	15	rad/s
K_{motor}	120	N
m	1.4	kg
c	1	-
l	0.25	m
I_x	0.03	$kg \cdot m^2$
I_y	0.03	$kg \cdot m^2$
I_z	0.04	$kg \cdot m^2$

TABLE II
THE RULE BASE FOR LEARNING-BASED FUZZY LQR CONTROLLER

		Error (e)				
		NL	NS	Z	PS	PL
Error Change ($\dot{e}$)	NL	Z	Z	Z	S	M
	NS	Z	Z	S	M	M
	Z	Z	S	M	M	M
	PS	S	S	M	B	B
	PL	M	M	M	B	B

B. Linearized Model

In this paper, using linearization to transform the nonlinear quadrotor helicopter model to be the kind of expression as Eq. (1) which is used for LQR controller design.

Simplified model can be obtained by assuming no yaw motion ($\psi = 0$) as well as the angles of pitch and roll motion are very small ($sin\phi \approx \phi \;\; and \;\; sin\theta \approx \theta$). Here, only the height motion model is illustrated as the state space form which is used in the simulation:

$$\begin{bmatrix} \dot{z} \\ \ddot{z} \end{bmatrix} = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} z \\ \dot{z} \end{bmatrix} + \begin{bmatrix} 0 \\ 1/m \end{bmatrix} u + \begin{bmatrix} 0 \\ -1 \end{bmatrix} g \quad (20)$$

IV. SIMULATION RESULTS

In this section, the above discussed learning-based fuzzy LQR control is simulated with both step response control and loss of partial control effectiveness performance of a quadrotor helicopter, and the simulation results of the proposed control technique will be presented for performance comparison with conventional LQR.

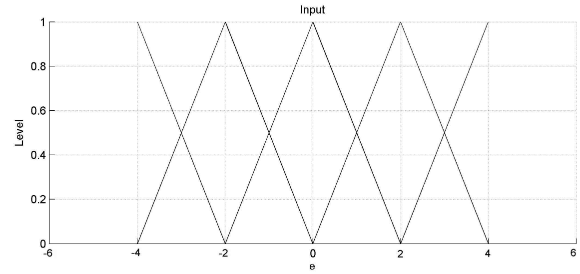

Fig. 2. Initial membership functions of error (e) for the learning-based fuzzy LQR before training.

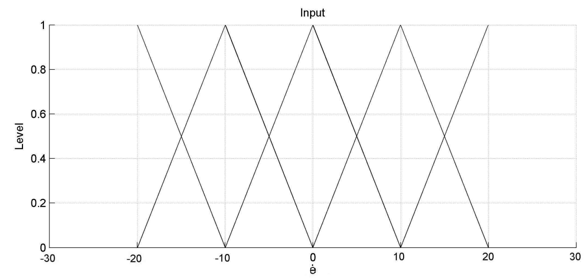

Fig. 3. Initial membership functions of derivative of error ($\dot{e}$) for the learning-based fuzzy LQR before training.

An illustrative example of the nonlinear system (equations in Eq. (17)) is chosen to evaluate the remarkable performance of the proposed method.

As shown in Table II, a fuzzy rule base with five membership functions for each of the two inputs and one output was defined. Where "NL" = "negative large", "NS" = "negative small", "Z" = "zero", "PS" = "positive small", and "PL" = "positive large" are chosen as fuzzy values for the inputs. "S" = "small", "M" = "medium", "B" = "big", and "Z" = "zero" are selected as fuzzy values for output. Fig. 2, Fig. 3, and Fig. 4 illustrate the initial membership functions and the shape sizes of error and derivative of this error for input, and output respectively. The range of the error for input was chosen as $[-4, \; 4]$ and the switch rate of this error has been selected as $[-20, \; 20]$. For fuzzy sets width of the output is $[-1, \; 1]$.

The base weighting matrix Q and R are the same for conventional LQR and learning-based fuzzy LQR. Their values are $Q = [0.25 \quad 69.44 \quad 0.00137 \quad 2500]$ and $R = 277.78$. The input constraints are set to be 0.1 for both of the two controllers.

A. Step Response Test

In the first set of experiments, the step response of two controllers are compared and shown in Fig. 8. Dash black line is the step input, solid green line is the response of conventional LQR, and dash red line is the proposed learning-based fuzzy-LQR response. Although the conventional LQR controller was capable of rising to the desired height, it was not able to eliminate the undesired overshoot. A signif-

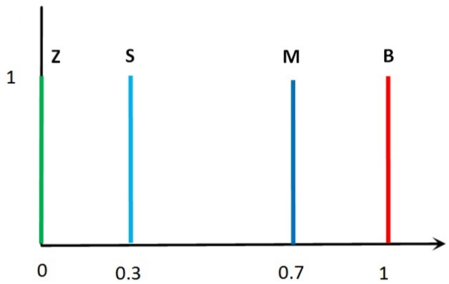

Fig. 4. Initial membership functions of output for the learning-based fuzzy LQR before training.

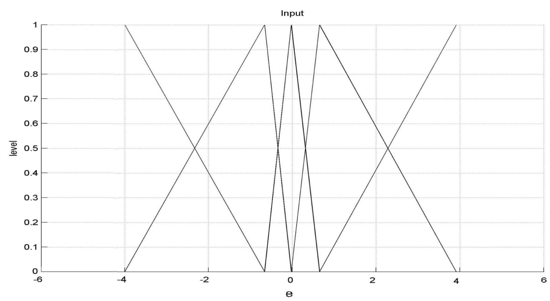

Fig. 5. Membership functions of error (e) for the learning-based fuzzy LQR after training.

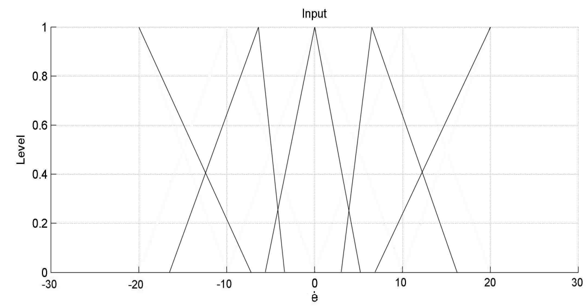

Fig. 6. Membership functions of derivative of error ($\dot{e}$) for the learning-based fuzzy LQR after training.

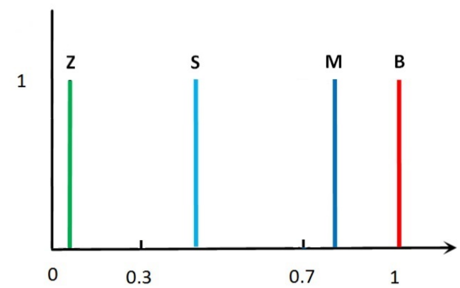

Fig. 7. Membership functions of output for the learning-based fuzzy LQR after training.

icant decrease in overshoot is seen with the proposed LQR controller and the settling time is shorten. The oscillating nature of conventional LQR response is completely removed in the proposed LQR controller response as well. Clearly, the proposed learning-based fuzzy-LQR controller shows better result than the conventional LQR. Fig. 5, Fig. 6 and Fig. 7 are the optimized results of the fuzzy LQR controller by using EKF. As these results show, the membership functions of the fuzzy LQR controller are automatically tunned on-line by EKF.

B. Loss of Partial Control Effectiveness Test

In the second set of experiments, the proposed control scheme was verified for its fault tolerant capability [13]. It is assumed that there is a fault (loss of partial control effectiveness of 20%) occurred in all four motors at 10th second. This kind of fault leads to a loss of altitude. Fig. 9 shows the performance of two controller's height control when control loss happened. As illustrated in the figure, although the conventional LQR controller maintains the desired height after the transient period induced by partial loss of control effectiveness, it is not satisfactory in the dynamic performance due to around 14% of tracking error at the moment of fault occurrence. Comparing with the result of the proposed controller, it reduced greatly the tracking error, There is only about 4% tracking error, and the system goes back to its desired position after around 2 seconds and the input is also in the acceptable range of the quadrotor helicopter's motors.

V. CONCLUSIONS AND FUTURE WORK

This paper addressed a novel online tuning fuzzy LQR control method which was based on the traditional LQR control combined with self-tuning fuzzy sets whose shapes of Mamdani fuzzy system are tuned on-line by using an EKF algorithm.

Simulations were carried out to evaluate the effectiveness of the proposed control method applied for height control of a quadrotor helicopter. The simulation results showed that the proposed on-line tuning fuzzy LQR controller could achieve good performance with respect to step reference input signal and in case of partial loss effectiveness of the motors. In addition, the proposed controller was compared with the conventional LQR controller to prove convincingly that the controller designed by fuzzy LQR methodology and EKF technique could satisfy the performance requirement.

Future work in this area could focus on the convergence of the Kalman filter, on-line model identification to yield more precise performance, the optimization of the fuzzy systems with other non-triangular membership functions.

ACKNOWLEDGMENT

This work is partially supported by the Natural Sciences and Engineering Research Council of Canada (NSERC), and China Scholarship Council (CSC).

REFERENCES

[1] R. G. Berstecher, R. Palm and H. D. Unbehauen, An adaptive fuzzy sliding-mode controller, *IEEE Trans. Industrial Electronics*, vol. 48, no. 4, pp. 18-31, 2001.

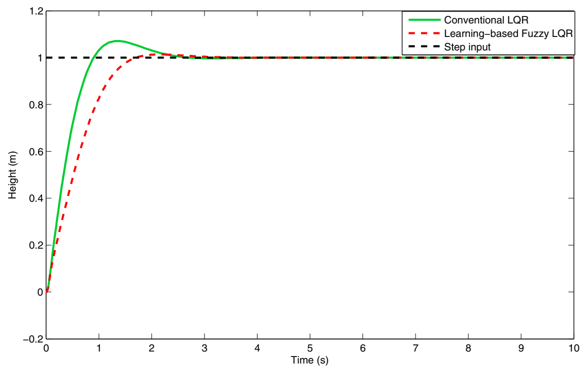

Fig. 8. Step response comparison between conventional and learning-based fuzzy LQR.

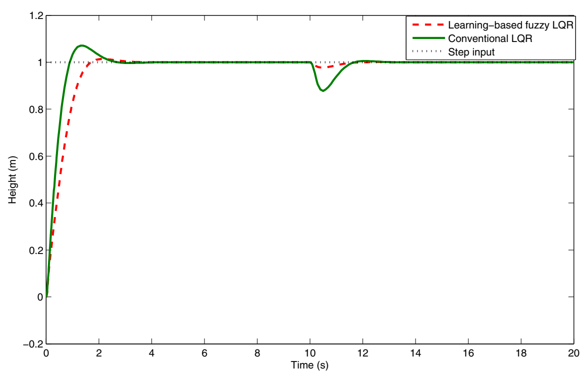

Fig. 9. Response performance comparison between conventional and learning-based fuzzy LQR when motors lost 20% control effectiveness.

[2] M. Sugeno and T. Yasukawa, A fuzzy-logic-based approach to qualitative modeling, *IEEE Trans. Fuzzy System*, vol. 1, no. 1, pp. 7-31, 1993.

[3] D. Simon, Sum normal optimization of fuzzy membership functions, *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, vol. 10, no. 4, pp. 363-384, 2002.

[4] C. W. Tao, J. S. Taur, and Y. C. Chen, Design of a parallel distributed fuzzy LQR controller for the twin rotor multi-input multi-output system, *International Journal of Uncertainty, Fuzzy Sets and Systems*, vol. 161, no. 15, pp. 2081-2103, 2010.

[5] A. Jimnez, B. M. Al-Hadithi and F. Mata, Extended Kalman filter for the estimation and fuzzy optimal control of Takagi-Sugeno model, *Fuzzy Controllers, Theory and Applications*, pp. 91-110, 2011.

[6] K. J. Astrm and B. Wittenmark, Computer-controlled systems, *Prentice Hall*, USA, 1997.

[7] A. Aswani, H. Gonzalez, S. S. Sastry, and C. Tomlin, Provably safe and robust learning-based model predictive control, *Automatica*, vol. 49, no. 5, pp. 1216-1226, 2013.

[8] R. Yazdanpanah, M. J. Mahjoob and E. Abbasi, Fuzzy LQR controller for heading control of an unmanned surface vessel, *International conference in electrical and electronics engineering*, Turkey, pp. 73-78, 2013.

[9] M. S. Grewal and A. P. Andrews, *Kalman filtering: theory and practice using MATLAB*, Wiley, 2011.

[10] L. C. Lai, C. C. Yang and C. J. Wu, Time-optimal control of a hovering quad-rotor helicopter, *Journal of Intelligent & Robotic Systems*, vol. 45, no. 2, pp. 115-135, 2006.

[11] M. Abdolhosseini, Y. M. Zhang and C. A. Rabbath, An efficient model predictive control scheme for an unmanned quadrotor helicopter. *Journal of Intelligent & Robotic Systems*, vol. 70, no. 1-4, pp. 27-38, 2013.

[12] K. K. Ahn and D. Q. Truong, Online tuning fuzzy PID controller using robust extended Kalman filter, *Journal of Process Control*, vol. 19, no. 6, pp. 1011-1023, 2009.

[13] Y. M. Zhang, and J. Jiang, Bibliographical review on reconfigurable fault-tolerant control systems, *Annual Reviews in Control*, vol. 32, no. 2, pp. 229-252. 2008.

[14] M. H. Amoozgar, A. Chamseddine and Y. M. Zhang, Fault-tolerant fuzzy gain scheduled PID for a quadrotor helicopter testbed in the presence of actuator faults, In *IFAC Conference on Advances in PID Control*, Brescia, Italy, 2012.

[15] K. T. Öner, E. Çetinsoy, M. Ünel, M. F. Akşit, I. Kandemir and K. Gülez, Dynamic model and control of a new quadrotor unmanned aerial vehicle with tilt-wing mechanism, *International Conference on Control, Automation, Robotics and Vision*, Paris, France, 2008.

[16] K. T. Öner, E. Çetinsoy, E. Sirimoglu, C. Hancer, T. Ayken and M. Ünel, LQR and SMC stabilization of a new unmanned aerial vehicle, *International Conference on Intelligent Control, Robotics, and Automation*, Venice, Italy, 2009.

[17] H. Jafari, M. Zareh, J. Roshanian and A. Nikkhah, An optimal guidance law applied to quadrotor using LQR method, *Transactions of the Japan Society for Aeronautical and Space Sciences*, vol. 53, no. 179, pp. 32-39, 2010.

[18] E. Reyes-Valeria, R. Enriquez-Caldera, S. Camacho-Lara and J. Guichard, LQR control for a quadrotor using unit quaternions: modeling and simulation, *International Conference on Electronics, Communications and Computing*, pp. 172-178, 2013.

[19] D. F. Bareiss and J. van den Berg, Reciprocal collision avoidance for quadrotor helicopters using LQR-Obstacles, *Twenty-Sixth AAAI Conference on Artificial Intelligence*, 2012.

[20] T. Nuchkrua and M. Parnichkun, Identification and optimal control of quadrotor, *International Journal of Science and Technology*, vol. 17, no. 4, pp. 36-53, 2012.

[21] T. Dierks and S. Jagannathan, Neural network output feedback control of a quadrotor UAV, *IEEE Conference on Decision and Control*, pp. 3633-3639, 2008.

[22] M. A. Olivares-Mendez, L. Mejias, P. Campoy and I. Mellado-Bataller, Cross-Entropy optimization for scaling factors of a fuzzy controller: a see-and-avoid approach for unmanned aerial systems, *Journal of Intelligent & Robotic Systems*, vol. 69, no. 1-4, pp. 189-205, 2013.

[23] M. A. Olivares-Mendez, P. Campoy, I. Mellado-Bataller and L. Mejias, See-and-avoid quadcopter using fuzzy control optimized by cross-entropy, *IEEE International Conference on Fuzzy Systems*, pp. 1-7, 2012.