

Modeling and Designing Fuzzy Membership Function based Neural Network Controller for Multi-Rotors Micro Aircraft Vehicle

Song Zhu¹, Zhixuan Wang¹, Chen Xiangjian^{1*} and Li Di²

^{*1}*Jiangsu University of Science and Technology, China
Email: cxj831209@163.com*

²*China Shipbuilding Industry corporation 723, China
Email: lee_di@163.com*

Abstract

For overcoming with the current MAV's poor flexibility, stability; lower lift, payload; coupling with the movements and motion changes when vertical taking off, landing and hovering, proposed the new Multi-rotors Micro Aircraft Vehicle (MAV). This paper provides the dynamic modeling of six degrees of freedom (6 DOF) of the Multi-rotors MAV. The derived model composed of translational and rotational subsystems is dynamically unstable; designing a neuro-fuzzy controller which is based on the fuzzy membership function based neural networks (FMFBNN) for autonomous Multi-rotors MAV. The FMFBNN controller combines the advantages of fuzzy logics and neural networks, such as inference capability and adoption of human operators' experience with fuzzy logics, and universal approximation and learning capability with neural networks, unlike other conventional control approaches; the present FMFBNN controller does not require off-line learning procedures, or human intervention to adjust parameters. On-line learning of the FMFBNN controller is achieved by using an inner-loop learning scheme. The simulated test flights indicate the capability of the FMFBNN controller in achieving the desired performance.

Keywords: *Micro Aircraft Vehicle, Fuzzy membership function based neural network*

1. Introduction

In the current work, Micro Aircraft Vehicle (MAV) is able to accomplish air observations and should be able to track an object on the pre-defined path, when it moves, or a way point following navigation towards a goal position. In order to design the proposed control system, some inherent properties of MAV are required. These are for example bounded system parameters, exerted disturbances, and skew-symmetry property. These requirements are considered in the design process.

Various advanced Micro Aircraft Vehicle (MAV) have been proposed recently. However the major limitation of helicopters' design is the need for extensive, and costly, maintenance for reliable operation. MAV rotorcraft is no exception. Simplifying the mechanical structure of such craft clearly produces logistical benefits. Multi-rotors vehicle is an alternative form of rotorcraft [1-5] which does not have the complicated swash plates and linkages found in conventional designs, and instead use varying rotor speeds to manoeuvre, the other advantage of the Multi-rotors vehicle configuration is its load capacity. Due to the great reduction of mechanical complexity and wear, it is expected that well-designed Multi-rotors will prove inherently more robust and reliable.

Meanwhile the MAV control systems have been proposed in the literature, such as sliding mode control, nonlinear control, adaptive control, neural network and fuzzy control. Conventional fuzzy control schemes require expert knowledge or

many cycles of trial-and-error to achieve the desired performance. In the case of neural network control, training time is unpredictable and neural networks may not be suitable for real-time control.

Conventional fuzzy control schemes require an expert's knowledge or many cycles of trial-and-error to achieve the desired performance. In neural network control, training time is unpredictable and neural networks may not be suitable for real-time control. Proposed neuro-fuzzy controller, called self-adaptive neuro-fuzzy inference system (SANFIS), for MAVs. The SANFIS controller can make fuzzy rules automatically with self-learning parameters. However, it requires learning the relationship between input and output using off-line learning schemes with input-output data generated by another control system.

This paper describes a neuro-fuzzy controller that was developed based on the first author's previous work on Fuzzy Membership Function-based Neural Network (FMFBNN) [7,8] and applied to control MAVs. Section 2 provides the dynamic modeling for the proposed Multi-rotors MAV. Section 3 describes the FMFBNN controller. Effectiveness of the FMFBNN controller was investigated and it was also compared with a non-repressor based adaptive controller by computer simulation. The simulation results are discussed in Section 4 before the conclusion which is drawn in Section 5.

2. Dynamic and Kinematical of Multi-rotors MAV

Mathematical dynamic models of flight behaviour are essential for good control design and analysis. The most basic model used consists only of rigid body dynamics with abstract force and torque actuators and no aerodynamics. The Multi-rotors vehicle is commonly represented as a rigid body mass with inertia and autogyroscopics, acted upon by gravity and control torques.

This section provides the specific model information of the Multi-rotors vehicle architecture starting from the generic 6 DOF rigid-body equation derived with the Newton-Euler formalism. Eq.1 shows the dynamic of the systems which can be rewritten in a system of the equations:

$$\left\{ \begin{array}{l} \ddot{\phi} = \frac{\sum M_x - \dot{\theta}\dot{\psi}(I_z - I_y)}{I_x} = \frac{U_{mx} + \dot{\theta}\dot{\psi}(I_z - I_y) - J_r\dot{\theta}\dot{\Omega}}{I_x} \\ \ddot{\theta} = \frac{\sum M_y - \dot{\phi}\dot{\psi}(I_x - I_z)}{I_y} = \frac{U_{my} + \dot{\phi}\dot{\psi}(I_x - I_z) - J_r\dot{\phi}\dot{\Omega}}{I_y} \\ \ddot{\psi} = \frac{\sum M_z - \dot{\phi}\dot{\theta}(I_y - I_z)}{I_z} = \frac{U_{mz} + \dot{\phi}\dot{\theta}(I_y - I_z)}{I_z} \\ \ddot{x} = \frac{S_{b-g}\sum F_{xb}}{m} = \frac{(\sin\theta\cos\phi\cos\psi + \sin\psi\sin\phi)U_x}{m} \\ \ddot{y} = \frac{S_{b-g}\sum F_{yb}}{m} = \frac{(\sin\theta\cos\phi\sin\psi - \cos\psi\sin\phi)U_y}{m} \\ \ddot{z} = \frac{S_{b-g}\sum F_{zb}}{m} = \frac{(\cos\theta\cos\psi)U_z}{m} - g \end{array} \right. \quad (1)$$

3. Controller Design

Consider the following N fuzzy relations with p input fuzzy variables:

$$R^m: \text{If } x_1 \text{ is } A_{n1}, x_2 \text{ is } A_{n2}, \dots \text{ and } x_p \text{ is } A_{np}, \text{ then } y \text{ is } B_n \quad (2)$$

Where $n=1,2,\dots,N$, x_j ($j=1,2,\dots,p$) is the j th input variable, u is the output fuzzy variable fuzzified with singleton membership function, and A_{ij} and B_i ($i=1,2,\dots,N$, and $j=1,2,\dots,p$) are input and output linguistic (fuzzy-set) values, respectively. If the center-of-gravity

defuzzification is applied, fuzzy relations in Eq.2 can be represented by fuzzy terms as follows:

$$F(x) = \sum_{j=1}^N \omega_j \Psi_j(x), \quad (3)$$

Where $\Psi_i(\cdot)$ is the fuzzy membership function and ω_i is the location of fuzzy singleton. Eq.3 is called FMFBNN. To reduce the computation load, the bases function of FMFBNN $\Psi_i(\cdot)$ which normally uses a triangular membership function of the input fuzzy variable for the i th fuzzy rule. The weight of FMFBNN ω_i is iteratively refined to minimize $\|f(x,t) - F(x,t)\|$ with the gradient descent method. An error function for FMFBNN is given by

$$E(x,t) = \frac{1}{2} (f(x,t) - \xi(x,t))^2 \quad (4)$$

And system output ξ can be represented as

$$\xi(x) = \Gamma(F(x)) = \Gamma(\sum_{i=1}^N \omega_i \Psi_i(x)) \quad (5)$$

Where $\Gamma(\cdot)$ represents the nonlinear dynamics from $F \in R^n$ to $\xi \in R^m$, the error function $E(t)$ with respect to the weights of the i th FMFBNN, ω_i can be obtained as

$$\begin{aligned} \frac{\partial E(x,t)}{\partial \omega_i(t)} &= -(f(x,t) - \xi(x,t)) \frac{\partial \xi(x,t)}{\partial \omega_i(t)} = -(f(x,t) - \xi(x,t)) \frac{\partial \xi(x,t)}{\partial F(x,t)} \frac{\partial F(x,t)}{\partial \omega_i(t)} \\ &= -(f(x,t) - \xi(x,t)) \frac{\partial \xi(x,t)}{\partial F(x,t)} \Psi_i(x,t) \end{aligned} \quad (6)$$

The next step is to get the derivative of $\xi(x,t)$ if change of $F(x,t)$ in a given time interval, we can assume that

$$\frac{\partial \xi(x,t)}{\partial F(x,t)} \approx \frac{\xi(x,t) - \xi(x,t-1)}{F(x,t) - F(x,t-1)} \quad (7)$$

Thus, the derivative of $E(t)$ with respect to $\omega_i(t)$ can be obtained as

$$\frac{\partial E(x,t)}{\partial \omega_i(x,t)} \approx -(f(x,t) - \xi(x,t)) \frac{\xi(x,t) - \xi(x,t-1)}{F(x,t) - F(x,t-1)} \Psi_i(x,t) \quad (8)$$

And the learning rule for adapting the weight of FMFBNN can be given as follows:

$$\omega_i(t+1) = \omega_i(t) + \eta (f(x,t) - \xi(x,t)) \frac{\xi(x,t) - \xi(x,t-1)}{F(x,t) - F(x,t-1)} \Psi_i(x,t) \quad (9)$$

And the learning rate parameter. It is noted that FMFBNN with Eq.9 would simplify the learning and retrieving procedure unlike the previous FMFBNN described in [7] where a Sigmoid function was used after defuzzification of fuzzy reasoning like a BP neural network. The overall diagram of FMFBNN with an unknown system is shown in Figure 2.

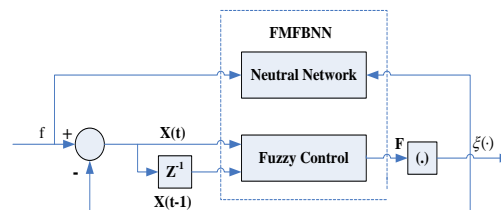


Figure 1. FMFBNN Controller

In order to regulate the system output to a desired output, more accurate control actions are needed near the set point of the desired output. Therefore, more fuzzy values must be placed near the set point. Since the fuzzy membership function for the THEN part is fuzzy singleton, its support is a single point in the center with a membership function.

4. Simulation

The Multi-rotors MAV is designed and built as a test-bed MAV at the Autonomous systems laboratory of the CIOMP. The Multi-rotors MAV has six thrusters. It is capable of maneuvering with six degrees of freedom (DOF). Each training and retrieving pattern consists of 12 inputs of state variables $x=[x, y, z, \phi, \theta, \psi]^T$, and change of errors, $\dot{x}=x(t)-x(t-1)$ and six outputs of forces and torques of position and orientation $y=[F_x, F_y, F_z, M_x, M_y, M_z]^T$. The error is defined as a difference between the desired value and the actual value in position and orientation of the vehicle.

Table 1 is an example of initial fuzzy singleton values with respect to the error and the change of error. The diagonal outputs are all zeros, and other outputs are roughly assigned as ± 0.5 and ± 1.0 , according to the distance from the diagonal. Even though determining fuzzy rules is not a big issue in FMFBNN design, sign for each output is important to get minimum performance with initial fuzzy rules: for instance, negative outputs in left upper side in the table, positive outputs in right lower side, and zeros in diagonal.

Table 1. Fuzzy Rules for FMFBNN Controller

$E \ \dot{E}$	NL	NM	NS	ZE	PS	PM	PL
NL	- 0.1	-0.1	- 0.1	- 0.1	- 0.5	-0.5	0.0
NM	- 0.1	-0.1	- 0.1	- 0.1	- 0.5	0.0	0.5
NS	- 0.1	-0.1	- 0.5	- 0.5	0.0	0.5	0.5
ZE	- 0.1	-0.5	- 0.5	0.0	0.5	0.5	1.0
PS	- 0.5	-0.5	0.0	0.5	0.5	1.0	1.0
PM	- 0.5	0.0	0.5	0.5	1.0	1.0	1.0
PL	0.0	0.5	0.5	1.0	1.0	1.0	1.0

The speed of the learning process is helped by using the following schemes. First, the training pattern are decomposed into six subsets, each of which consists of the position and orientation error, the change of error, the force and torque in each axis. The coupling effects between subsystems will be compensated by neural network. Second, the assumed $f(\cdot), \xi(\cdot), \Psi_i(\cdot)$ in Eq.8 as the constant during the learning of each data pattern, which is called “inner-loop learning”. The calculation of inner-loop learning is performed as a foreground task while control inputs are calculated in the time-critical background task such as timer interrupt routine. With the learning rate $\eta=0.01$, during the simulation, the Multi-rotors vehicle was controlled to follow the straight line of desired path in the body coordinates; Fig.3 shows the actual trajectory by the neuro-fuzzy controller with the initial normalized fuzzy rule shown in Table 1. It is noted that the FMFBNN controller perform well than the normal adaptive controller as time goes by with continuous learning. Fig.4 shows the actual trajectory by the normal adaptive controller. To investigate the

performance with learning, the overall performance in terms of errors along with the desired trajectory was measured by the Eq.10.

$$T = \frac{1}{S} \sum_{k=1}^S \sqrt{\sum_{i=1}^6 e_i(k)^2} \quad (10)$$

Where S is the total number of samples during each learning cycle, Figure 5 shows the value of each learning cycle which is getting smaller with more learning cycles.

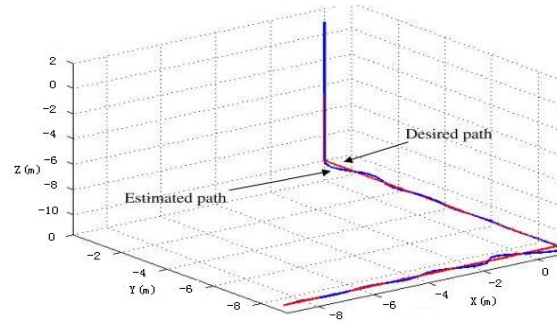


Figure 2. The Trajectory Comparison between the Desired and one with FMFBNN Controller

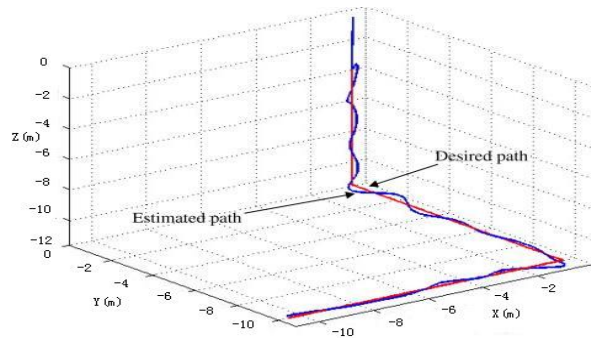


Figure 3. The Trajectory Comparison between the Desired and with the Normal Adaptive Controller

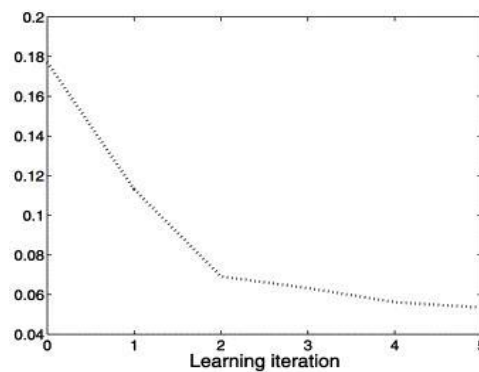


Figure 4. The Performance of Eq.1 with Initial Fuzzy Rules in Table 1

5. Conclusion

In this paper, a new neuro-fuzzy controller based on FMFBNN and its application to the Multi-rotors vehicle was discussed. Unlike conventional neural network control or fuzzy control, the presented controller does not require a large number of training data in advance. The FMFBNN controller uses on-line learning that is fast enough to realize real-

time control. Simulation results show that the controller is robust with fuzzy rules designed and performs well in the laboratory.

As the FMFBNN controller does not require more information about the vehicle system and is very attractive especially for Multi-rotors MAV control since the vehicle is a very complex, highly nonlinear, time-varying dynamic system including gyro dynamic parameter uncertainties and unknown disturbance such as wind. Especially, when the vehicle performance gets degraded during the operation in air, due to changes in the system or its environment, the proposed controller could easily adapt to the changes and provide desired performance without human operator's intervention. Future research includes experimental study of the proposed controller.

Acknowledgment

This paper is supported by National Natural Science Found of China No.61502211.

References

- [1] W. Ren and R. W. Beard, "Trajectory tracking for unmanned air vehicles with velocity and heading rate constrains", IEEE Transactions on control systems technology, vol. 12, no. 5, (2004), pp. 706-716.
- [2] G. Xun and B. Yue, "Hex-Rotor aircraft and its autonomous flight control system", Optics and Precision Engineering, vol. 20, no. 11, (2012), pp. 2450-2458.
- [3] J. S. R. Jang, "ANFIS: adaptive-network-based fuzzy inference system", IEEE Transactions on Systems, Man, and Cybernetics, vol. 23, no. 3, (1993), pp. 665-684.
- [4] Y. Lin and G. A. Cunningham, "A new approach to fuzzy-neural system modeling", IEEE Transactions on Fuzzy Systems, vol. 3, no. 2, (1995), pp. 190-198.
- [5] C. F. Juang and C. T. Lin, "An online self-constructing neural fuzzy inference network and its applications", IEEE Transactions on Fuzzy Systems, vol. 8, no. 1, (1998), pp. 12-32.
- [6] S. Q. Wu and M. J. Er, "Dynamic fuzzy neural networks-a novel approach to function approximation", IEEE Transactions on Systems, Man, and Cybernetics, Part B, vol. 30, no. 2, (2000), pp. 358-364.
- [7] S. Q. Wu, M. J. Er and Y. Gao, "A fast approach for automatic generation of fuzzy rules by generalized dynamic fuzzy neural networks", IEEE Transactions on FLzzy Systems, vol. 9, no. 4, (2001), pp. 578-594.
- [8] Y. Gao, M. J. Er and S. Yang, "Adaptive control of robot manipulators using fuzzy neural networks", IEEE Transactions on Industrial Electronics, vol. 48, no. 6, (2001), pp. 1274-1278.
- [9] M. J. Er and Y. Gao, "Robust adaptive control of robot manipulators using generalized fuzzy neural networks", IEEE Transactions on Industrial Electronics, vol. 60, no. 3, (2003), pp. 620-628.
- [10] Y. Gao and M. J. Er, "Online adaptive fuzzy neural identification and control of a class of MIMO nonlinear systems", IEEE Transactions on Fuzzy Systems, vol. 11, no. 4, (2003), pp. 462-477.