PARAMETERS OPTIMIZATION FOR UNMANNED AERIAL VEHICLE CONTROL

Juraj Gvoth, Michal Blaho, Tatiana Mudráková

Institute of Robotics and Cybernetics

Faculty of Electrical Engineering and Information Technology

Slovak University of Technology

Ilkovičova 3, 812 19 Bratislava, Slovak Republic

Abstract

Nowadays, the hottest topic in the field of control belongs to the Unmanned Aerial Vehicle UAV. These devices are not controlled directly from their cockpits and can therefore be deployed precisely in the dangerous areas. From a perspective view the UAV device belongs to the MIMO systems which means that several inputs affects several outputs. This paper is devoted to the UAV device control design in consideration to the speed control. Control of the device has been designed using fuzzy logic that closely reproduce the control logic of the human interaction. Fuzzy logic parameters were optimized using genetic algorithm.

1 Introduction

Quadcopter is a device similar to a helicopter, but in contrast to the helicopter it uses four rotors rotated upwards and distributed at the corners for its flight. It is a complex system, which can be mathematically described [1] and modeled in Matlab Simulink environment [2]. Despite its complexity it can be controled without the knowledge of the mathematical model by experienced operators. Knowledge of operators can be transformed by means of artificial intelligence (fuzzy logic) to the control system [3]. Fuzzy logic was successfully used in the control of various UAV devices [4,5]. Improved solution for the control system can be achieved by using optimization algorithms, for example, genetic algorithms [6]. This article describes the design of quadcopter simulation model in MATLAB Simulink, the expansion of the neglected components, fuzzy controller design and its optimization using genetic algorithms.

2 Mathematical model of quadcopter

The mathematical model describes the quadcopter dynamics in the simplified sense. If we would like to model all the effects on the behavior of quadcopter during the flight, the mathematical model would be considerably more complicated. Simulation will be also complicated, which would then become difficult for a computer to calculate. The mathematical model is taken from the literature [7]. In the Figure 1 the basic structure of quadcopter can be seen and also can be found the world coordinate system, the coordinate quadcopter system, angular velocity directions of each rotor, and the torque and tension forces generated by rotors.

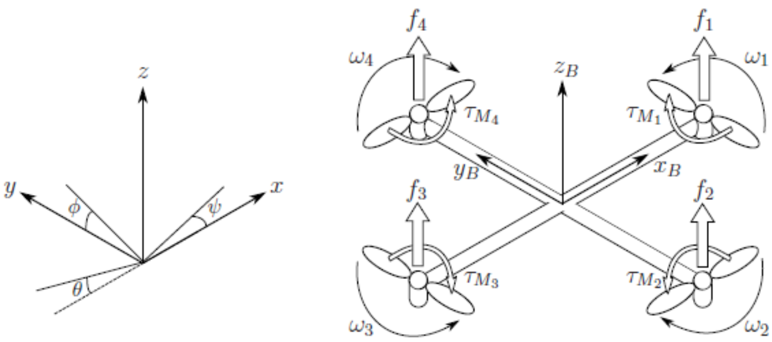

Figure 1: The principal quadcopter view in two coordinate system[7]

The quadcopter position in the world coordinate system is defined by the vector ξ and the vector of angular displacement **η**.

$$\boldsymbol{\xi} = \begin{bmatrix} x \\ y \\ z \end{bmatrix} \tag{1}$$

$$\eta = \begin{bmatrix} \phi \\ \theta \\ \psi \end{bmatrix} \tag{2}$$

Quadcopter displacement in the world coordinate system is defined by the Euler angles, where **ϕ** je roll angle determines the rotation around x-axis, **θ** pitch angle represents the rotation of the quadcopter around the y-axis and jaw angle **ψ** around the z-axis. Vector **ν** contains the linear and angular position vectors.

$$\nu = \begin{bmatrix} p \\ q \\ r \end{bmatrix} \tag{3}$$

The rotation matrix **R** from the quadcopter coordinate system to the world coordinate system,

$$R = \begin{bmatrix} C_\psi C_\theta & C_\psi S_\theta S_\phi - S_\psi C_\phi & C_\psi S_\theta C_\phi + S_\psi S_\phi \\ S_\psi C_\theta & S_\psi S_\theta S_\phi + C_\psi C_\phi & S_\psi S_\theta C_\phi - C_\psi S_\phi \\ -S_\theta & C_\theta S_\phi & C_\theta C_\phi \end{bmatrix} \tag{4}$$

where is $S_x=sin(x)$ and $C_x=cos(x)$. The rotation matrix **R** is orthogonal, thus apply $\mathbf{R}^{-1}=\mathbf{R}^{T}$ which is the rotation matrix from the world coordinate system to the quadcopter coordinate system. The transformation matrix $\boldsymbol{W_\eta}$ for angular velocities from the world coordinate system to the quadcopter coordinate system and $\boldsymbol{W_\eta^{-1}}$ vice versa.

$$\nu = W_\eta \dot{\eta}$$

$$\begin{bmatrix} p \\ q \\ r \end{bmatrix} = \begin{bmatrix} 1 & 0 & -S_\theta \\ 0 & C_\phi & C_\theta S_\phi \\ 0 & -S_\phi & C_\theta C_\phi \end{bmatrix} \begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} \tag{5}$$

$$\dot{\eta} = W_\eta^{-1} \nu$$

$$\begin{bmatrix} \dot{\phi} \\ \dot{\theta} \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} 1 & S_\phi T_\theta & C_\phi T_\theta \\ 0 & C_\phi & -S_\phi \\ 0 & S_\phi / C_\theta & C_\phi / C_\theta \end{bmatrix} \begin{bmatrix} p \\ q \\ r \end{bmatrix} \tag{6}$$

The quadcopter assume to have symmetric structure with the four rotors in the x-axis and y-axis, the inertia matrix is diagonal matrix **I** where $I_{xx} = I_{yy}$.

$$I = \begin{bmatrix} I_{xx} & 0 & 0 \\ 0 & I_{yy} & 0 \\ 0 & 0 & I_{zz} \end{bmatrix} \tag{7}$$

Each of the motors create force $\mathbf{f_i}$ and torque $\boldsymbol{\tau_{Mi}}$ around the rotor axis,

$$f_i = k\omega_i^2 \tag{8}$$

$$\tau_{M_i} = b\omega_i^2 + I_M \dot{\omega}_i \tag{9}$$

where $\dot{\omega}_i$ is often very small a therefore can be neglected. The sum of forces created by each rotor generates the tension force **T** in the direction of z-axis the quadcopter coordinate system.

$$T = \sum_{i=1}^{4} f_i = k \sum_{i=1}^{4} \omega_i^2$$

$$T_B = \begin{bmatrix} 0 \\ 0 \\ T \end{bmatrix} \tag{10}$$

The final vector of torques consisting of torques which directions responds to the direction of quadcopter coordinate system,

$$\tau_B = \begin{bmatrix} \tau_\phi \\ \tau_\theta \\ \tau_\psi \end{bmatrix} = \begin{bmatrix} lk(-\omega_2^2 + \omega_4^2) \\ lk(-\omega_1^2 + \omega_3^2) \\ \sum_{i=1}^{4} \tau_{M_i} \end{bmatrix} \tag{11}$$

where l is the distance between the rotor and the center of mass of the quadcopter and k is the constant for the force created by the rotor. The quadcopter r is assumed to be rigid body and thus Newton-Euler equations can be used to describe its dynamics. at first it will be used equations describing the actual position of the entity in the world coordinate system. In the equation (12) the condition of equilibrium state is recorded in the quadcopter coordinate system, where the force required for the acceleration of mass **m** and the centrifugal force $\boldsymbol{\nu} \times (\mathrm{m}V_B)$ have to be equal to the gravity $\mathbf{R^T G}$ and the total thrust of the rotors $\mathbf{T_B}$

$$m\dot{V}_B + \nu \times (mV_B) = R^T G + T_B \tag{12}$$

In the world coordinate system, the centrifugal force is nullified. Thus only the gravitational force and the magnitude and direction of the thrust are contributing in the acceleration of the quadcopter.

But because we need the equation with respect to the world coordinate system, it is necessary to multiply the right side of the equation with rotation matrix from which we get the equation (12).

$$\xi = m\ddot{\xi} = G + RT_B \begin{bmatrix} x \\ y \\ z \end{bmatrix} \tag{13}$$

After substituting the status description (14) is obtained, which contains the first motion equations of quadcopter

$$\begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} = -g \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} + \frac{T}{m} \begin{bmatrix} C_\psi S_\theta C_\phi + S_\psi S_\phi \\ S_\psi S_\theta C_\phi - C_\psi S_\phi \\ C_\theta C_\phi \end{bmatrix} \tag{14}$$

In the quadcopter coordinate system the equation (15) is applied, as well. Where the angular acceleration of the inertia $\mathbf{I\dot{\nu}}$, the centripetal forces $\boldsymbol{\nu} \times (\mathbf{I\nu})$ and the gyroscopic forces $\boldsymbol{\Gamma}$ are equal to the external torque $\boldsymbol{\tau}$.

$$I\dot{\nu} + \nu \times (I\nu) + \Gamma = \tau$$

$$\dot{\nu} = I^{-1} \left(- \begin{bmatrix} p \\ q \\ r \end{bmatrix} \times \begin{bmatrix} I_{xx} p \\ I_{yy} q \\ I_{zz} r \end{bmatrix} - I_r \begin{bmatrix} p \\ q \\ r \end{bmatrix} \times \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \omega_\Gamma + \tau \right)$$

$$\begin{bmatrix} \dot{p} \\ \dot{q} \\ \dot{r} \end{bmatrix} = \begin{bmatrix} (I_{yy} - I_{zz}) qr / I_{xx} \\ (I_{zz} - I_{xx}) pr / I_{yy} \\ (I_{xx} - I_{yy}) pq / I_{zz} \end{bmatrix} - I_r \begin{bmatrix} q / I_{xx} \\ -p / I_{yy} \\ 0 \end{bmatrix} \omega_\Gamma + \begin{bmatrix} \tau_\phi / I_{xx} \\ \tau_\theta / I_{xx} \\ \tau_\psi / I_{xx} \end{bmatrix} \tag{15}$$

Equation $\omega_\Gamma = \omega_1 - \omega_2 + \omega_3 - \omega_4$ is applied. In order to work with the angle displacement of quadcopter in the world coordinate system, it is necessary to use the transformation equation (6). To ensure a more realistic model, the member that representing the force exerted by air resistance needs to be add to the equation (14). It represents a diagonal matrix vector multiplication with an aerodynamic drag coefficient in the individual axes and linear velocities of quadcopter in the world coordinate system.

$$\begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{z} \end{bmatrix} = -g \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} + \frac{T}{m} \begin{bmatrix} C_\psi S_\theta C_\phi + S_\psi S_\phi \\ S_\psi S_\theta C_\phi - C_\psi S_\phi \\ C_\theta C_\phi \end{bmatrix} - \frac{1}{m} \begin{bmatrix} A_x & 0 & 0 \\ 0 & A_y & 0 \\ 0 & 0 & A_z \end{bmatrix} \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{z} \end{bmatrix} \tag{16}$$

In a similar way, it is able to add further aerodynamic effects, such as vibration of the propeller blades, or the flow of the air. But this would have resulted in considerably more complex mathematical model.

2.1 Model extension

Motor dynamics and their control was in the quadcopter model neglected. It was therefore necessary to find parameters of RC brush-less DC motor, which is usually used for quadcopters in the practice. Parameters of such motor was defined accordingly

Table 1: Parameters for BLDC

Parameter	*Value*	*Unit*
K_E	$6.3*10^{-3}$	V s / rad^{-1}
K_M	$6.3*10^{-3}$	N m A^{-1}
R	0.001	Ω

This equation describes dynamics of such motor, similar to DC motor

$$\dot{\Omega}_n = -\frac{K_E K_M}{R I_M}\Omega_n - \frac{b}{I_M}\Omega_n^2 + \frac{K_M}{R I_M}v_n \qquad (17)$$

PI controllers were designed to these motors. With such motors it has been achieved fast and stable start-up in the vicinity of the operating point, which is the value 620 rad / s, when is the quadcopter able to hover in place. The "tune" function of the PID controller block in MATLAB Simulink environment was used on parameters design. The final parameters are P = 0.071 and I = 4585. Subsequently, the entire BLDC connection with PID were encapsulated into one subsystem, which was therefore inserted just before the angular velocity inputs into the original model. It is necessary to add noise into the model superposed to the values of the angular velocities of each motor to have mentioned model little bit closer to the real model. The block "White Noise" in Simulink serves to this purpose. This block provides the option for setting the "key" to generate white noise, called "seed". To ensure the uniqueness of the values sequence at the output of the block, it was as the key chosen current time by calling the function "clock".

3 Model verification

The model in the block diagram has been verified in the Simulink environment. Quadcopter constants and coefficients used in the simulation are in the Table 2.

Table 2: PARAMETERS FOR THE SIMULATION

Parameter	*Value*	*Unit*
g	9.81	m/s^2
m	0.468	kg
l	0.225	m
k	$2.980*10^{-6}$	-
b	$1.140*10^{-7}$	-
I_M	$3.357*10^{-5}$	kg m^2
I_{xx}	$4.856*10^{-3}$	kg m^2
I_{yy}	$4.856*10^{-3}$	kg m^2
I_{zz}	$8.801*10^{-3}$	kg m^2
A_x	0.25	kg/s
A_y	0.25	kg/s
A_z	0.25	kg/s

The simulation will issue from the state, when the rotors already have a value of angular velocity that the pulling force generated by them is equal to the gravitational force and thus the quadcopter will be in a stable position. During the first 0.2 seconds speed of all four rotors has been increased by the step change that causes movement of the quadcopter in the positive z direction. Opposite, at the time of 0.2 to 0.4 seconds the speed of all four rotors were reduced and from the time 0.4s were returned to the original value by a step change. Similarly, the angle φ (Roll) changes with fourth rotor speed increase and simultaneously second rotor speed decrease in the time of 0.5 to 0.7 seconds. Increase of angle φ was stopped by the time 0.7 to 0.9 seconds because the fourth rotor speed was reduced and the second rotor speed was increased. This change of the angle will reflects the quadcopter movement in the negative y-axis direction and in the negative z-axis position change direction. To achieve the angle θ (pitch) change, it was necessary in the time of 1 to 1.2 seconds increase the third rotor speed by step response and simultaneously to decrease the second rotor speed. Comparing the previous angle, the increase of the angle was also in this case stopped at the time of 1.2 to 1.4 seconds by the rotor speed decrease. At this point quadcopter starts to move in the positive x-axis direction and it even faster starts to sink on the z-axis. The angle ψ (Yaw) was last modified by increasing the first and third rotor speed and decreasing the second and fourth rotor speed at the time of 1.5 to 7.1 seconds. To prevent this angle not to accumulate, it was also required at the time of 1.5 to 1.7 seconds to reduce the first and third rotor speed and increase the second and fourth rotor speed. The time responses of the rotors speed, position, and rotation angle of the quadcopter body in the global coordinate system are displayed in the Figure 2.

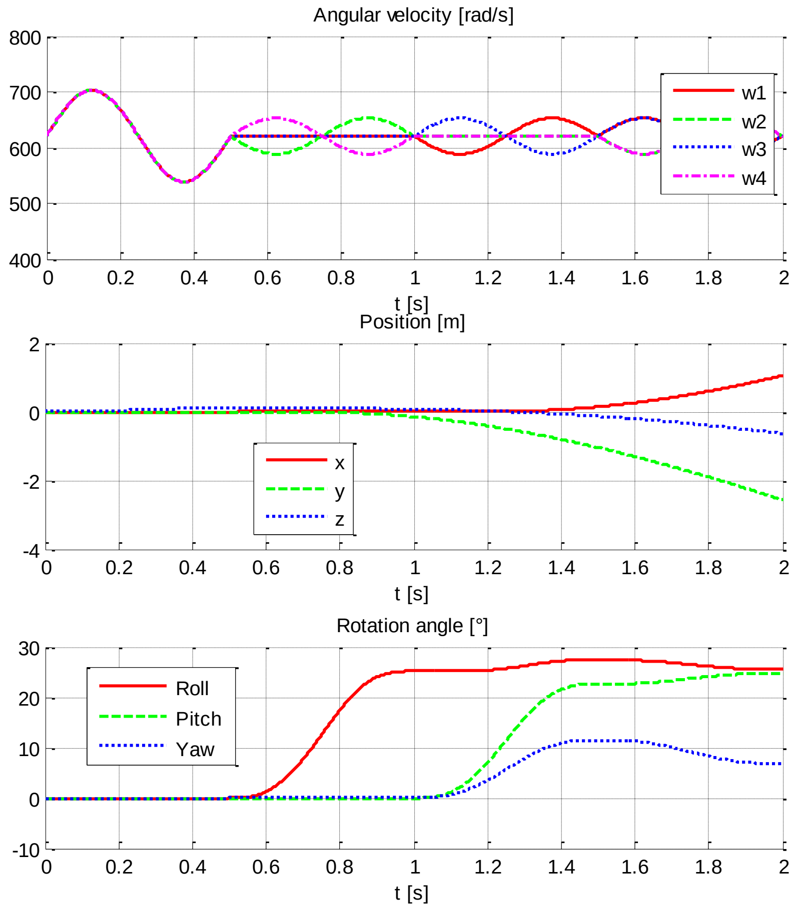

Figure 2: Time responses of the quadcopter angular velocities, position and rotation angle

4 Fuzzy control

The control design is the result of articles study [3,4]. Firstly, it was necessary to select a control structure that is appropriate considering the type of process. Due to the complexity and order of the system cascade connection structure was chosen. Internal secondary circuit will form the control of the quadcopter rotation angles and the primary external circuit will manage the translational movement control of the whole quadcopter. Controllers have been selected as fuzzy PD. Moreover, before control design it was necessary to create a block for data preprocessing. It was also necessary to create vectors, which will contain the position errors of the given states and of these states derivations. Block scheme can be seen in the figure below.

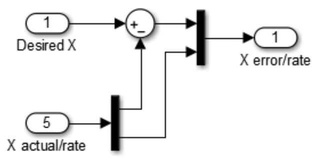

Figure 3: Pre-processing block diagram

On the same principle also works all other quadcopter states that need to be processed for control. But inner circuit output gives only information about which direction it needs to swing in order to achieve the desired rotation angle (and consequently the desired position). Therefore, it is still necessary to provide so-called block. "Post-processing", that sends setpoints for the respective rotors

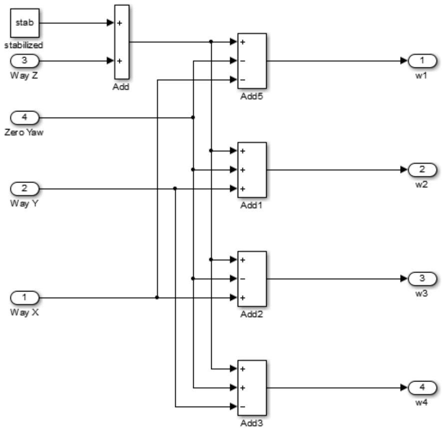

Figure 4: Post-procesing block diagram

Po štúdiu vedeckých článkov zaoberajúcich sa návrhom takého typu riadenia bolo usúdené, že samotný regulátor bude pre každý vstup obsahovať len tri funkcie príslušnosti, ktoré rozlíšia, či je hodnota kladná, záporná alebo rovná nule. To isté bude platiť pre výstup, povie nám len či máme ísť niektorým smerom, alebo už nemusíme robiť nič. Tvary funkcii príslušností sú rovnaké pre vstupné aj výstupné premenné líšia sa len v parametroch ako je na obrázku 8 v nasledujúcej kapitole. Báza pravidiel nachádzajúca sa v každom regulátore je na nasledujúcom obrázku

After studying scientific articles dealing with the design of this type of control, it was considered that the controller will contain for each input only three membership functions, which distinguish whether the value is positive, negative or zero. The same will stand for the output - it tells us only which direction to go, or to do nothing. The shapes of membership functions are the same for input and output variables, they differ only in the parameters, as shown in the Figure 8 of the next chapter. Base rules, situated in the each controller, is in the Figure 5.

		e		
		N	**Z**	**P**
dy	**N**	P	P	Z
	Z	P	Z	N
	P	Z	N	N

Figure 5: Rule base

All ranges of membership functions are in the range [-1, 1]. The actual scaling is provided at the inlet and output of the controller. As shown in the following Figure 6, it can be seen that the signal is at first limited and afterwards converted to the range [-1, 1]. And consequently on the output is the signal re-scaled again, if necessary.

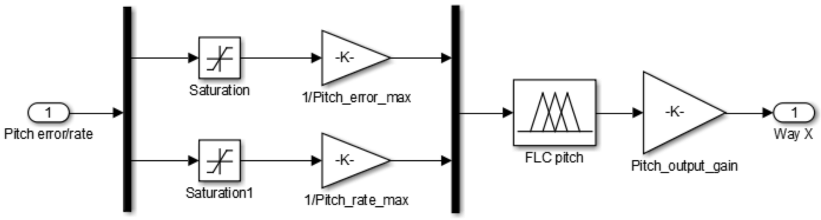

Figure 6: Block diagram of fuzzy controller with scaling its inputs and outputs

The actual circuits are connected in the basic block scheme called FLC (Fuzzy Logic Controller) and are connected according to the following scheme.

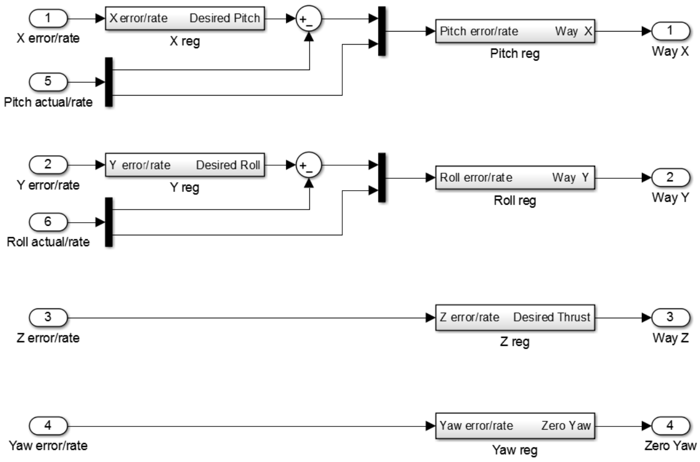

Figure 7: Link between internal and external control circuit

5 Parameter optimization

Stable solution and accurate quadcopter movement in space is possible to achieve with progressive adjustment of parameters entered by skilled operator. These parameters were optimized with some help of implementation and the use of genetic algorithm. It was sought such parameters that could help to speed up the quadcopter movement while preserving the steadiness of movement. Accent was placed on the speed of movement, therefore PD controller was used. Its disadvantage is the position error, which can be for certain tasks neglected. Important parameters in the controller structures are scaled inputs, outputs and internal distribution of membership sets. While setting ranges for inputs or outputs are simple, the internal structure already requires already editing * .fis files which involve all data describing the fuzzy controllers settings. First, we chose a system of shapes modification of membership functions as shown in the Figure 8.

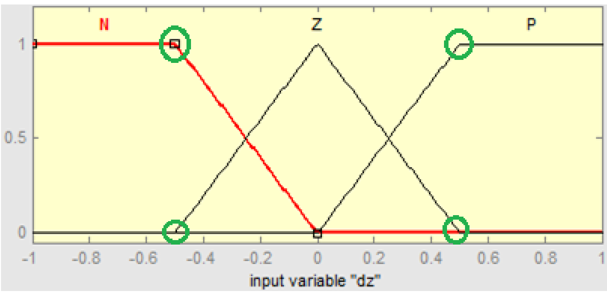

Figure 8: Symmetrical arrangement of membership functions

Points shown in the Figure 8, are symmetrical distributed around the zero point, indeed around values [-0.5; 0.5]. Thus, any change have to also occur in order to preserve this symmetry. The same is applied to other inputs and outputs. For this purpose the special function was created. Input to this function consists of the original structure and three values of the distances from zero point in absolute value, namely the error input, the change of speed and the output. The controller was designed to move in the direction of x-axis and y-axis with the same values of parameters (the difference is only in the sign). Similarly, controllers have also been designed for the rotation around the x-axis and y-axis. Therefore, they could join together and have been changed with shared variables as XY controller, or PR (Pitch and Roll). The ontrollers for movement in the z-axis and rotation around the z-axis (Yaw angle) remained separated. Thus we get for each controller six parameters that will be set using genetic algorithm. Objective function is used in the form

$$J = \sum_{k=1}^{N_{sim}} e_k + \alpha dy_k \tag{13}$$

with α value that was progressively adjusted to achieve fast and stable control.

6 Results

After optimizing the control parameters the scaling values of input and output have been changed, but mainly the distribution of fuzzy sets in the fuzzy controllers. From the Figure 9 and 10 can be seen that the greatest change occurred in the distribution of fuzzy sets in the output and the cover sets indicating directions widened, to which the quadcopter should sway or move.

Changes can be also seen at the maximum sets constriction at the inputs for the angle Yaw controller that reflects when the angle is already zero, or no longer changed. Position controller in the x-axis and y-axis significantly broadens the scope of central sets for the input of the position error and also reduced the set mean for the input speed of variable change. On the contrary, it is with a angle Pitch and Roll controllers, where except the sets change of the output membership functions, occurred no marked changes in the input set.

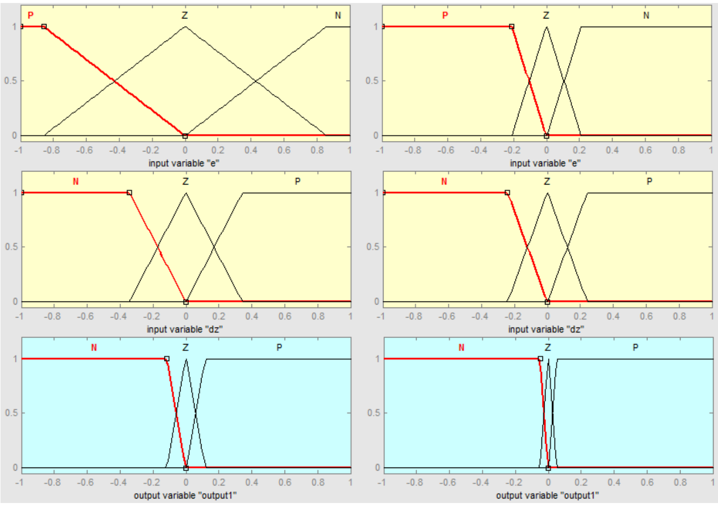

Figure 9: Changes in the assembly separation of the membership function –XY controller (on the left) and Z (on the right)

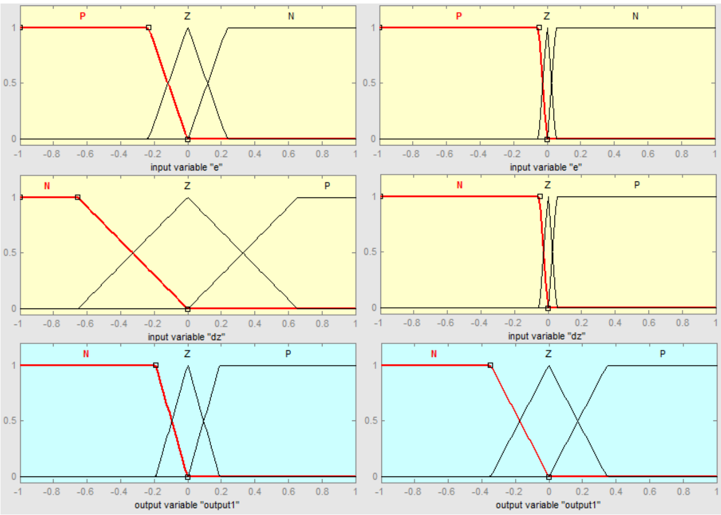

Figure 10: Periodic signal generator, block scheme

6.1 Transfer to the desired position

At first unit step for all directions in the space was tested. From the steady state in a coordinate system [0, 0, 0] (in meters) quadcopter has been sent to the point [1, 1, 1] (in meters). Figure 11 shows that the optimized parameters speeded up the quadcopter transfer in the space. At the same time, however, the dynamics of movement in the space has been changed. Although the speed of the device is higher, but it takes longer to get to the final desired state in the last centimeters.

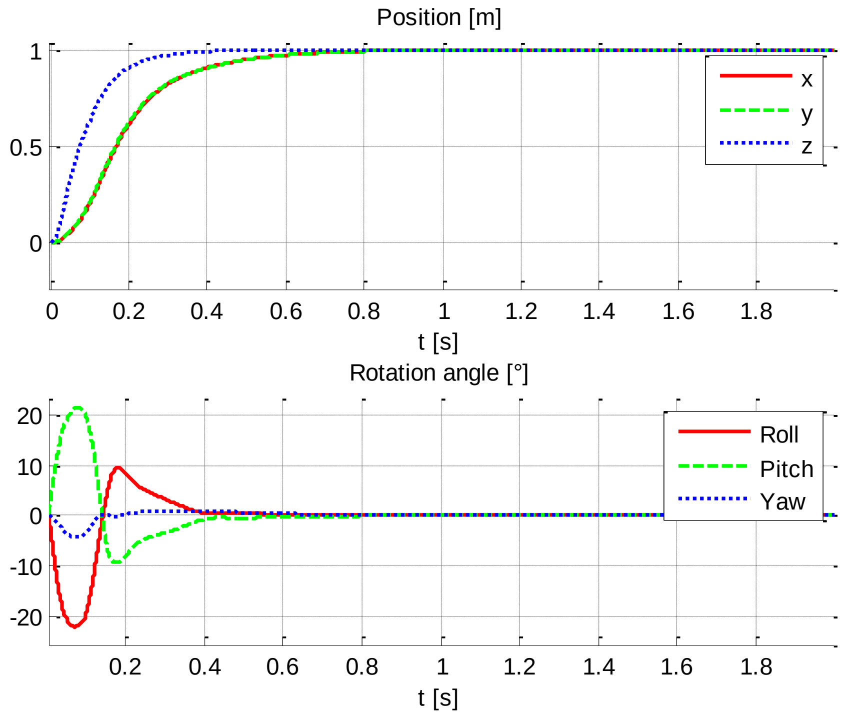

Figure 11: Quadcopter reaction on the step unit in the space

6.2 Transfer to the desired position considering the wind

Another test verified the stability of the individual control with noise, that simulates the effect of the wind on the quadcopter, to the controller inputs. In the Figure 12 can be seen that the quadcopter position curve in the space was markedly not changed, but the angles are not stabilized, nor in the case if the object has reached the desired position.

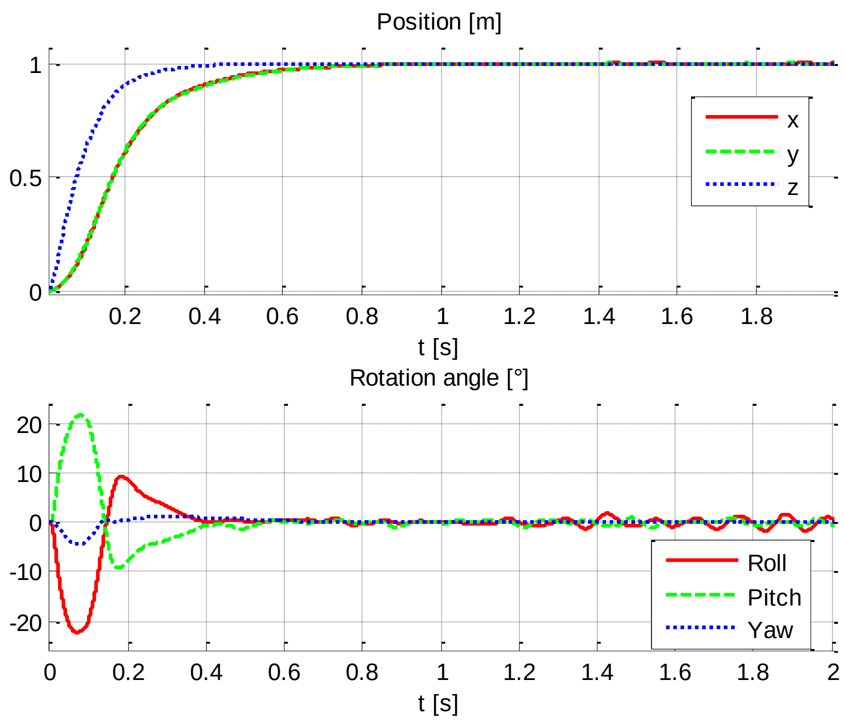

Figure 12: Quadcopter reaction on the step unit in the space

7 Conclusion

This article describes the design of quadcopter model control. Based on mathematical relationships from the literature [1] simulation model in MATLAB Simulink environment was created. This model was extended to the impact of motors and the environment. For quadcopter control was used fuzzy logic. Fuzzy uses decision making process closest to human. Based on the input and the combination of rules, decisions can be implemented without the need of mathematical model knowledge, which would otherwise have to be included in the process control design.

Control uses the cascade of internal and external circuit. Internal circuit provides quadcopter misalignment in desired directions. While the external circuit, on the basis of a comparison of the desired and actual quadcopter positions, determines how to swing it to achieve the control goal - its position. Optimization of the controller values was realized by genetic algorithm, focusing on the period of regulation. During the motion tracking came the permanent control deviation due to the absence of I controlling component. Anyway, this control is still able to deliver precise movements of the set-point and thus may find its application in motion, where does not matter moderate control deviation.

Acknowledgement

This paper was supported by the Scientific Grant Agency of the Ministry of Education, Science, Research and Sport of the Slovak Republic under the contract No. VEGA-1/1256/12.

References

[1] T. Luukonen. *Modelling and control of quadcopter*, School of Science, Espoo, August 22, 2011, http://www.sal.tkk.fi/publications/pdf-files/eluu11.pdf

[2] P. Karban. *Výpočty a simulace v programech Matlab a Simulink*. Computer Press, 2007, ISBN: 80-2511-448-1

[3] Mathworks. Fuzzy Logic Toolbox – Users Guide. http://www.mathworks.com/help/pdf_doc/fuzzy/fuzzy.pdf

[4] S. A. Raza, W. Gueaieb. *Intelligent Flight Control of an Autonomous Quadrotor*, Motion Control, Federico Casolo (Ed.), ISBN: 978-953-7619-55-8, InTech 2010

[5] D. Chaoyang. *Networked flexible spacecraft attitude maneuver based on adaptive fuzzy sliding mode control*, 2009 School of Aerospace Scienceand Engineering, Beihang University, China

[6] I. Sekaj. *Evolučné výpočty a ich využitie v praxi*. Bratislava: IRIS, 2005. 157 s. ISBN 80-89018-87-4.

Michal Blaho
michal.blaho@stuba.sk

Tatiana Mudráková
Tatiana.mudrakova@stuba.sk