https://orcid.org/0000-0001-7381-9461

UDC 519.769

T.H. HABIBBAYLI[*,**]

FORMATION OF THE QUADCOPTER FLIGHT PATH UNDER OVERLAND MONITORING USING NEURO-FUZZY MODELING METHODS

[*]Institute of Control Systems of Azerbaijan National Academy of Science, Baku, Azerbaijan
[**]Institute of Information Technology of Azerbaijan National Academy of Science, Baku, Azerbaijan

Анотація. *Окремі території, де ведеться господарська діяльність, характеризуються наявністю гірських і лісових масивів. Для інформаційної підтримки розвитку інфраструктури та сільського господарства на цих територіях у деяких випадках необхідний наземний моніторинг за допомогою безпілотних технологій. У зв'язку з цим запропоновано алгоритм формування тривимірної траєкторії руху квадрокоптера під час наземного пілотування в гірсько-лісовому ландшафті, який передбачає автономне маневрування для подолання можливих перешкод. Як базову модель пропонується використовувати систему нечіткого висновку з вхідними характеристиками у вигляді лінгвістичних змінних, що відображають нечіткі сектори простору, в межах яких вербально інтерпретуються наявність перешкод і відстань до них, тобто у вигляді термінів відповідних вхідних лінгвістичних змінних. Подолання перешкод передбачається виконувати на основі нечітких висновків запропонованої системи, сформульованих як терми вихідних лінгвістичних змінних, які відображають зміни кута повороту в горизонтальній площині, висоту польоту та швидкість польоту квадрокоптера. У даній статті аналізуються результати поведінки моделі для різних сценаріїв термів вхідних лінгвістичних змінних. Для оперативного формування траєкторії польоту квадрокоптера також пропонується використовувати засоби нейромережевого моделювання. Завдяки своїй здатності адаптуватися до нових умов і вимог, модель нейронної мережі може стати важливою ланкою в забезпеченні автономного польоту квадрокоптера в умовах наземного моніторингу. Для відповідного навчання тришарової прямої нейронної мережі використовується достатньо велика кількість сценаріїв поведінки квадрокоптера, які були згенерованою системою нечіткого висновку щодо подолання можливих перешкод у п'яти секторах опитування.*

Ключові слова: *квадрокоптер, траєкторія польоту, наземний моніторинг, нечіткий набір, нечіткий висновок, нейронна мережа.*

Abstract. *Some territories where economic activity is carried out are characterized by the presence of mountaini and forests. To provide information support for the development of infrastructure and agriculture in these areas, in some cases, there is required overland monitoring with unmanned technologies. In this regard, an algorithm for the formation of a 3D trajectory of a quadcopter during overland piloting in a mountainous and wooded landscape is proposed, which implies autonomous maneuvering to overcome possible obstacles. As a basic model, it is proposed to use a fuzzy inference system with input characteristics in the form of linguistic variables that reflect fuzzy sectors of space, within which the presence of obstacles and the distance to them are interpreted verbally, i.e., in the form of terms of corresponding input linguistic variables. Overcoming obstacles is supposed to be performed on the basis of fuzzy conclusions of the proposed system, formulated as terms of output linguistic variables which reflect changes in the angle of rotation in the horizontal plane, flight altitude, and traverse speed of the quadcopter. The paper analyzes the results of the model behavior for different scenarios of the terms of the input linguistic variables. For the operational formation of the quadcopter flight path, it is also proposed to use neural network modeling tools. Due to its ability to adapt to new conditions and requirements, the neural network model can become an important tool in ensuring the autonomous flight of a quadcopter under overland monitoring. For appropriate training of the three-layer feedforward neural network, a sufficiently large number of quadcopter behavior scenarios are used, which were generated by the fuzzy inference system relative to the overcoming of possible obstacles in five sectors of the survey.*

Keywords: *quadcopter, flight path, overland monitoring, fuzzy set, fuzzy inference, neural network.*

DOI: 10.34121/1028-9763-2022-3-97-107

1. Introduction

The current level of unmanned technologies actualizes the use of drones for overland monitoring of infrastructure and agriculture, characterized by large mountainous and forest areas. For overland monitoring of various objects, it is preferable to use rotary-wing drones (quadcopters) [1] which are the most popular among researchers despite their own limitations. The advantage of the quadcopter is that each of its four propellers provides raising force and high flight stability. Nevertheless, as a rather complex technical device, a quadcopter is characterized by flight dynamics that are difficult to formalize due to its "sensitivity" to the influence of external factors [1, 2]. Therefore, solving the problem of controlling a quadcopter for overland monitoring in the presence of natural obstacles in a mountainous and wooded landscape and hard-to-reach places is interesting both from scientific and practical points of view. In this regard, the technological solutions of DJI are of particular interest which produces compact quadcopters with intelligent flight modes [3].

2. Problem statement

In the absence of necessary information support and, as a result, the impossibility of constructing an accurate mathematical model, the application of fuzzy logic methods in quadcopter control is specified by the capabilities, which provide for compiling heuristic knowledge and using intuitive data. Currently, fuzzy approaches to solving problems of automatic control are considered in two aspects. The first is related to the creation of a classifier of situations that forms the goals and objectives of the functioning of a dynamic system. The second approach involves direct fuzzy regulation of the variables of the control object. Based on these prerequisites, the importance and actuality of developing software for the autonomous movement of the quadcopter under overland piloting in mountainous and wooded areas, including automatic maneuvering to bypass possible obstacles, become obvious. In the context of the foregoing, an algorithm is proposed for creating a "smart" obstacle avoidance system for quadcopter autopiloting, based on the combined use of the fuzzy inference system and a multilayer feedforward neural network.

The aim of the paper is to develop a neuro-fuzzy approach to the formation of the quadcopter flight path under overland monitoring.

3. Fuzzy inference system for quadcopter flight path regulation

The DJI Mavic 2 quadcopter is equipped with six sensors for detecting obstacles in all directions, which ensures high-quality drone performance even in the most difficult situations. Thanks to the Flight Autonomy autopilot system, all data are transmitted and processed permanently in real time. In [4], an algorithm for forming a flight path of a quadcopter was proposed using a fuzzy inference system based on expert and empirical data analysis. It provides overland autopiloting of a quadcopter equipped with obstacle detection sensors in five frontal viewing sectors (see Fig. 1).

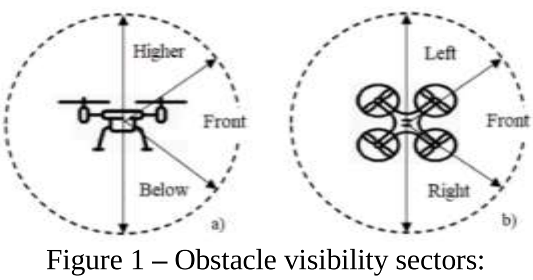

Figure 1 – Obstacle visibility sectors:
a) side view; b) top view

To form the schedule for overland autopiloting of the quadcopter in five directions, a bounded set of logically consistent rules is considered in the form of the following information fragments:

d_1: "If any obstacle is not detected on the flight path of the quadcopter or it is too far away, then there is no need to change direction, height and to reduce speed”;

d_2: “If the sensor detects an obstacle at a medium distance along the flight path of the quadcopter and the sector on the left is free, then it is necessary to lose velocity to an average value and slightly turn to the left without changing the height”;

d_3: “If the sensor detects an obstacle at a close distance along the flight path of the quadcopter and the sector on the left is free, then it is necessary to lose velocity to a minimum and sharply turn to the left without changing the height”;

d_4: “If the sensor detects an obstacle at a medium distance along the flight path of the quadcopter, there is also an obstacle in the left sector at a not remote distance and the sector on the right is free, then it is necessary to lose velocity to an average value and slightly turn to the right without changing the height”;

d_5: “If the sensor detects an obstacle at a close distance along the flight path of the quadcopter, there is also an obstacle in the left sector at a not remote distance and the sector on the right is free, then it is necessary to lose velocity to a minimum and sharply turn to the right without changing the height”;

d_6: “If the sensor detects an obstacle at an average distance along the flight path of the quadcopter, there are also obstacles in the left and right sectors at a not remote distance, and the upper sector is free, then it is necessary to lose velocity to an average value and slightly increase the flight altitude (pitch-up) without yaw”;

d_7: “If the sensor detects an obstacle at a close distance along the flight path of the quadcopter, there are also obstacles in the left and right sectors at a not remote distance, and the upper sector is free, then it is necessary to lose velocity to a minimum and sharply increase the flight altitude without yaw”;

d_8: “If the sensor detects an obstacle at an average distance along the flight path of the quadcopter, there are also obstacles in the left, right and upper sectors at a non-remote distance, and the lower sector is free, then it is necessary to lose velocity to an average value and slightly reduce the flight altitude (dive) without yaw”;

d_9: “If the sensor detects an obstacle at a close distance along the flight path of the quadcopter, there are also obstacles in the left, right and upper sectors at a not remote distance, and the lower sector is free, then it is necessary to lose velocity to a minimum and sharply reduce the flight altitude without yaw”;

d_{10}: “If the sensor detects an obstacle at an average distance along the flight path, and an obstacle is also detected at an average distance to the left, obstacles are detected at a not remote distance to the right, below and above, then it is necessary to lose velocity to an average value while maintaining the course and flight altitude”;

d_{11}: “If the sensor detects an obstacle at a close distance along the flight path of the quadcopter, and an obstacle is also detected at an average distance to the left, obstacles are detected at a not remote distance to the right, below and above, then it is necessary to lose velocity to a minimum and sharply turn to the left without changing the height”;

d_{12}: “If the sensor detects an obstacle at a medium distance along the flight path of the quadcopter, and an obstacle is detected a close distance to the left, an obstacle is detected at an average distance to the right, and obstacles are detected at a not remote distance from below and above along the course, then it is necessary to lose velocity to an average value and sharply turn to the left without changing the course and height”;

d_{13}: “If the sensor detects an obstacle at a close distance along the flight path of the quadcopter, and an obstacle is also detected at a close distance to the left, an obstacle is detected at an average distance to the right, and obstacles are detected from below and above along the course at a not remote distance, then it is necessary to lose velocity to a minimum and sharply turn to the right without changing the height”;

d_{14}: “If the sensor detects an obstacle at a medium distance along the flight path, an obstacle is detected at a close distance to the left, an obstacle is also detected at a close distance to the right, an obstacle is detected at an average distance from above, and an obstacle is detected at a not remote distance from below, then it is necessary to lose velocity to an average value and sharply turn to the right without changing the course and height”;

d_{15}: “If the sensor detects an obstacle at a close distance along the flight path of the quadcopter, the obstacles are detected at an average distance to the left and the right, as well as an obstacle is detected at a not remote distance from above, then it is necessary to lose velocity to a minimum and sharply increase flight altitude without yaw”;

d_{16}: “If the sensor detects an obstacle at a medium distance along the flight path, obstacles are detected at a close distance to the left, right and higher along the course, and an obstacle is detected at an average distance below the course, then it is necessary to lose velocity to an average value without changing the course and height”;

d_{17}: “If the sensor detects the obstacles at a close distance along the flight path of the quadcopter, as well as to the left, right and above, however, an obstacle is detected at an average distance below the course, then it is necessary to lose velocity to a minimum and sharply reduce the flight altitude without yaw”;

d_{18}: “If an obstacle is detected at a medium distance along the flight path, obstacles are detected at a close distance to the left, right, above and below the course, then it is necessary to lose velocity to an average value without changing the course and height”;

d_{19}: “If in all sectors of the view the detected obstacles are at a close distance, then it is necessary to lose velocity to a minimum and sharply turn to the left without changing the height”.

Maneuvering to the left (or to the right, which is also appropriate) from the frontal impasse caused by the presence of obstacles in all five sectors of view (see rule d19), the quadcopter continues to move and, thereby, creates a new flight path for itself in accordance with the regulations established by rules $d_1 \div d_{19}$. Thus, analysis of all possible scenarios of collision with obstacles made it possible to form a complete set of linguistic variables (see Table 1) and rules for forming a fuzzy inference system (FIS) that regulates the behavior of a quadcopter during overland piloting.

The corresponding FIS is formed by the following rules in a symbolic form:

d_1: $(x_1=X_{11}) \Rightarrow (y_1=Y_{11})$ & $(y_2=Y_{23})$ & $(y_3=Y_{33})$;
d_2: $(x_1=X_{12})$ & $(x_2=X_{21}) \Rightarrow (y_1=Y_{12})$ & $(y_2=Y_{22})$ & $(y_3=Y_{33})$;
d_3: $(x_1=X_{13})$ & $(x_2=X_{21}) \Rightarrow (y_1=Y_{13})$ & $(y_2=Y_{21})$ & $(y_3=Y_{33})$;
d_4: $(x_1=X_{12})$ & $(x_2=\neg X_{21})$ & $(x_3=X_{31}) \Rightarrow (y_1=Y_{12})$ & $(y_2=Y_{24})$ & $(y_3=Y_{33})$;
d_5: $(x_1=X_{13})$ & $(x_2=\neg X_{21})$ & $(x_3=X_{31}) \Rightarrow (y_1=Y_{13})$ & $(y_2=Y_{25})$ & $(y_3=Y_{33})$;
d_6: $(x_1=X_{12})$ & $(x_2=\neg X_{21})$ & $(x_3=\neg X_{31})$ & $(x_4=X_{41}) \Rightarrow (y_1=Y_{12})$ & $(y_2=Y_{23})$ & $(y_3=Y_{34})$;
d_7: $(x_1=X_{13})$ & $(x_2=\neg X_{21})$ & $(x_3=\neg X_{31})$ & $(x_4=X_{41}) \Rightarrow (y_1=Y_{13})$ & $(y_2=Y_{23})$ & $(y_3=Y_{35})$;
d_8: $(x_1=X_{12})$ & $(x_2=\neg X_{21})$& $(x_3=\neg X_{31})$& $(x_4=\neg X_{41})$& $(x_5=X_{51}) \Rightarrow (y_1=Y_{12})$ & $(y_2=Y_{23})$ & $(y_3=Y_{32})$;
d_9: $(x_1=X_{13})$&$(x_2=\neg X_{21})$& $(x_3=\neg X_{31})$ & $(x_4=\neg X_{41})$ & $(x_5=X_{51}) \Rightarrow (y_1=Y_{13})$ & $(y_2=Y_{23})$ & $(y_3=Y_{31})$;
d_{10}: $(x_1=X_{12})$ & $(x_2=X_{22})$ & $(x_3=\neg X_{31})$& $(x_4=\neg X_{41})$& $(x_5=\neg X_{51}) \Rightarrow (y_1=Y_{12})$& $(y_2=Y_{23})$ & $(y_3=Y_{33})$;
d_{11}: $(x_1=X_{13})$ & $(x_2=X_{22})$& $(x_3=\neg X_{31})$& $(x_4=\neg X_{41})$& $(x_5=\neg X_{51}) \Rightarrow (y_1=Y_{13})$ & $(y_2=Y_{21})$ & $(y_3=Y_{33})$;
d_{12}: $(x_1=X_{12})$ & $(x_2=X_{23})$& $(x_3=X_{32})$ & $(x_4=\neg X_{41})$ & $(x_5=\neg X_{51}) \Rightarrow (y_1=Y_{12})$ & $(y_2=Y_{23})$ & $(y_3=Y_{33})$;
d_{13}: $(x_1=X_{13})$ &$(x_2=X_{23})$ & $(x_3=X_{32})$ & $(x_4=\neg X_{41})$ & $(x_5=\neg X_{51}) \Rightarrow (y_1=Y_{13})$ & $(y_2=Y_{25})$ & $(y_3=Y_{33})$;
d_{14}: $(x_1=X_{12})$ & $(x_2=X_{23})$ & $(x_3=X_{33})$ & $(x_4=X_{42})$ & $(x_5=\neg X_{51}) \Rightarrow (y_1=Y_{12})$ & $(y_2=Y_{23})$ & $(y_3=Y_{33})$;
d_{15}: $(x_1=X_{13})$ & $(x_2=X_{23})$ & $(x_3=X_{33})$ & $(x_4=X_{42})$ & $(x_5=\neg X_{51}) \Rightarrow (y_1=Y_{13})$ & $(y_2=Y_{23})$ & $(y_3=Y_{35})$;
d_{16}: $(x_1=X_{12})$ & $(x_2=X_{23})$ & $(x_3=X_{33})$ & $(x_4=X_{43})$ & $(x_5=X_{52}) \Rightarrow (y_1=Y_{12})$ & $(y_2=Y_{23})$ & $(y_3=Y_{33})$;
d_{17}: $(x_1=X_{13})$ & $(x_2=X_{23})$ & $(x_3=X_{33})$ & $(x_4=X_{43})$ & $(x_5=X_{52}) \Rightarrow (y_1=Y_{13})$ & $(y_2=Y_{23})$ & $(y_3=Y_{31})$;
d_{18}: $(x_1=X_{12})$ & $(x_2=X_{23})$ & $(x_3=X_{33})$ & $(x_4=X_{43})$ & $(x_5=X_{53}) \Rightarrow (y_1=Y_{12})$ & $(y_2=Y_{23})$ & $(y_3=Y_{33})$;

d_{19}: $(x_1=X_{13})$ & $(x_2=X_{23})$ & $(x_3=X_{33})$ & $(x_4=X_{43})$ & $(x_5=X_{53}) \Rightarrow (y_1=Y_{13})$ & $(y_2=Y_{21})$ & $(y_3=Y_{33})$.

Table 1 – Input and output linguistic variables of the FIS and their terms

Inputs			
Symbol	Variable name	Universe	Term set
x_1	The remoteness of the obstacle in the direction of flight	[0; 1]	$\{X_{11}$=SIGNIFICANT, X_{12}=AVERAGE, X_{13}=INSIGNIFICANT$\}$
x_2	The remoteness of the obstacle to the left of the direction	[0; 1]	$\{X_{21}$=SIGNIFICANT, X_{22}=AVERAGE, X_{23}=INSIGNIFICANT$\}$
x_3	The remoteness of the obstacle to the right of the direction	[0; 1]	$\{X_{31}$=SIGNIFICANT, X_{32}=AVERAGE, X_{33}=INSIGNIFICANT$\}$
x_4	The remoteness of the obstacle above the direction	[0; 1]	$\{X_{41}$=SIGNIFICANT, X_{42}=AVERAGE, X_{43}=INSIGNIFICANT$\}$
x_5	The remoteness of the obstacle below the direction	[0; 1]	$\{X_{51}$=SIGNIFICANT, X_{52}=AVERAGE, X_{53}=INSIGNIFICANT$\}$
Outputs			
y_1	Airspeed	[0; 1]	$\{Y_{11}$=FULL, Y_{12}=AVERAGE, Y_{13}=ZERO$\}$
y_2	Yaw	[-0,5; 0,5]	$\{Y_{21}$=SHARPLY TO THE LEFT, Y_{22}=SLIGHTLY TO THE LEFT, Y_{23}=IS ABSENT, Y_{24}=SLIGHTLY TO THE RIGHT, Y_{25}=SHARPLY TO THE RIGHT$\}$
y_3	Pitch	[-0,5; 0,5]	$\{Y_{31}$=SHARPLY UP, Y_{32}=SLIGHTLY UP, Y_{33}=IS ABSENT, Y_{34}=SLIGHTLY DOWN, Y_{35}=SHARPLY DOWN$\}$

To implement the FIS that provides autopiloting of the quadcopter according to the formalized schedule, it is necessary to reflect the introduced linguistic variables to the set of their corresponding real numbers by setting membership functions, i.e., fuzzify the terms of all input and output linguistic variables. For this purpose, the advantage of one element over another relative to the property of the given fuzzy subset of the discrete universe $U=\{0,\ 0{,}1,\ 0{,}2,\ \ldots,\ 1\}$ is estimated. According to Saaty's 9-point scale [5], for example, the matrix of paired comparisons of the elements of the universe for identifying the membership function of the fuzzy set "UNSIGNIFICANT" as one of the terms of the input linguistic variable "Remoteness of obstacles" (see Table 1) is presented as follows:

$$X_{13}=\left[\begin{array}{c|ccccccccccc}
\text{Remoteness} & 0 & 0{,}1 & 0{,}2 & 0{,}3 & 0{,}4 & 0{,}5 & 0{,}6 & 0{,}7 & 0{,}8 & 0{,}9 & 1 \\ \hline
0 & 1 & 1 & 1 & 3 & 7 & 9 & 9 & 9 & 9 & 9 & 9 \\
0{,}1 & 1 & 1 & 1 & 3 & 7 & 9 & 9 & 9 & 9 & 9 & 9 \\
0{,}2 & 1 & 1 & 1 & 3 & 5 & 9 & 9 & 9 & 9 & 9 & 9 \\
0{,}3 & 1/3 & 1/3 & 1/3 & 1 & 3 & 5 & 5 & 5 & 7 & 9 & 9 \\
0{,}4 & 1/7 & 1/7 & 1/5 & 1/3 & 1 & 1 & 3 & 3 & 5 & 7 & 9 \\
0{,}5 & 1/9 & 1/9 & 1/9 & 1/5 & 1 & 1 & 1 & 3 & 3 & 5 & 7 \\
0{,}6 & 1/9 & 1/9 & 1/9 & 1/5 & 1/3 & 1 & 1 & 1 & 1 & 3 & 5 \\
0{,}7 & 1/9 & 1/9 & 1/9 & 1/5 & 1/3 & 1/3 & 1 & 1 & 1 & 1 & 3 \\
0{,}8 & 1/9 & 1/9 & 1/9 & 1/7 & 1/5 & 1/3 & 1 & 1 & 1 & 1 & 1 \\
0{,}9 & 1/9 & 1/9 & 1/9 & 1/9 & 1/7 & 1/5 & 1/3 & 1 & 1 & 1 & 1 \\
1 & 1/9 & 1/9 & 1/9 & 1/9 & 1/9 & 1/7 & 1/5 & 1/3 & 1 & 1 & 1
\end{array}\right],$$

where, if the preference of the i-th number over the j-th has one of the presented values, the assessment of the preference of the j-th number over the i-th has the inverse value $(i,\ j = 0, 1, 2, ..., 10)$.

Finding the eigenvector $(\mu_0,\ \mu_1, ...,\ \mu_{10})$ from the equation $(X_{13} - \lambda E)\mu = 0$ $(X_{13} - \lambda E)$, there are obtained the values of the membership function $\mu(u_i)$ of the fuzzy set X_{13}, which are summarized in Table 2. Each such membership function can be approximated as given in a tabular form and obtain a graphical representation of the corresponding fuzzy set.

Table 2 – The values of the membership function of the fuzzy set UNSIGNIFICANT

Remoteness	0	0,1	0,2	0,3	0,4	0,5	0,6	0,7	0,8	0,9	1
$\mu(u_i)$	1	1	1	0,9865	0,0113	0,0098	0,0064	0,0035	0,0021	0,0008	0,0002

In MATLAB\FIS notation, all membership functions were established empirically. In particular, the input and output characteristics of the model are shown in Fig. 2 and Fig. 3, respectively.

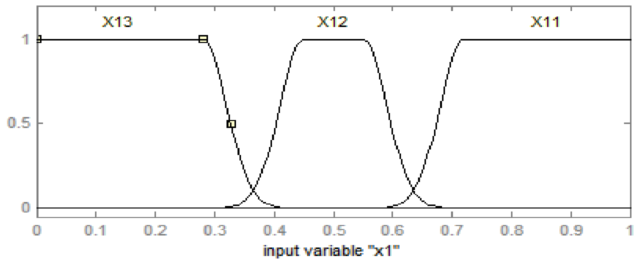

Figure 2 – Terms of input linguistic variable “The remoteness of the obstacle in the course”

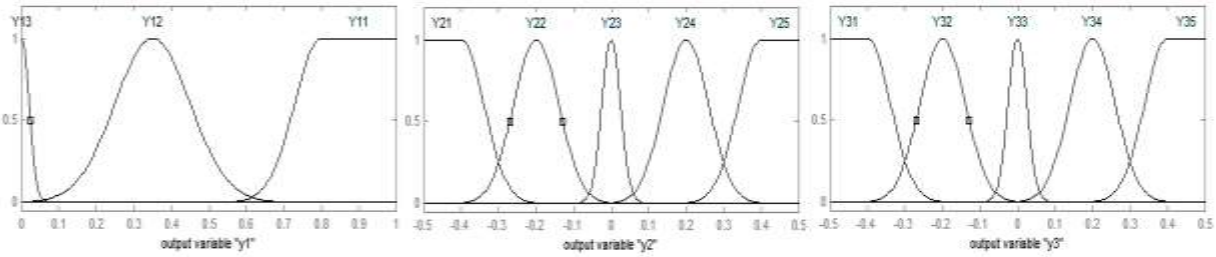

Figure 3 – Terms of output linguistic variables: y_1 – airspeed, y_2 – yaw, y_3 – pitch

As a result, using the interactive window of the graphical interface of the MATLAB\FIS editor (see Fig. 4), it was possible to generate the products, which are summarized in Table 3, and different flight scenarios provided in Table 4.

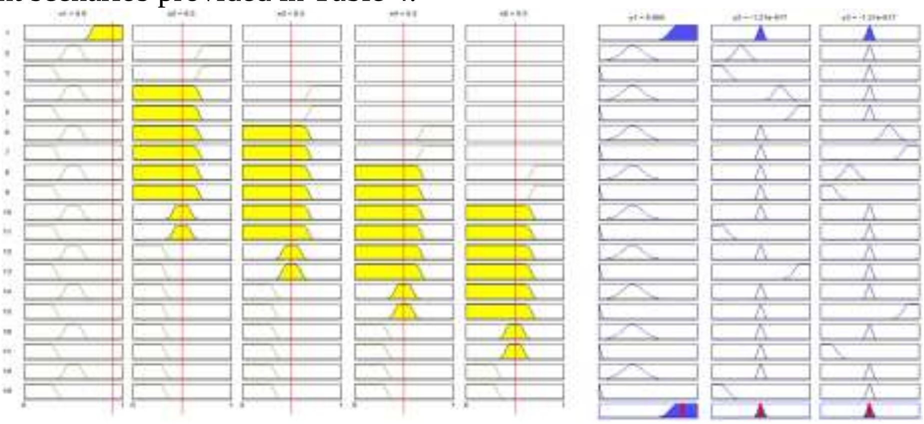

Figure 4 – Graphical interface of the MATLAB\FIS editor for viewing rules

Table 3 – The products generated by MATLAB\FIS editor

№	Inputs					Outputs		
	x_1	x_2	x_3	x_4	x_5	y_1	y_2	y_3
1	0,95	0,50*	0,50	0,50	0,50	0,855	0	0
2	0,50	0,95	0,50	0,50	0,50	0,350	-0,200	0
3	0,10	0,90	0,50	0,50	0,50	0,013	-0,411	0
4	0,50	0,50	0,90	0,50	0,50	0,350	0,200	0
5	0,10	0,50	0,90	0,50	0,50	0,013	0,411	0
6	0,50	0,50	0,50	0,90	0,50	0,350	0	0,200
7	0,10	0,50	0,50	0,90	0,50	0,013	0	0,411
8	0,50	0,50	0,50	0,50	0,90	0,350	0	-0,200
9	0,10	0,50	0,50	0,50	0,90	0,013	0	-0,411
10	0,50	0,50	0,40	0,40	0,40	0,350	0	0
11	0,10	0,50	0,50	0,50	0,50	0,013	-0,411	0
12	0,50	0,10	0,50	0,50	0,50	0,350	0	0
13	0,10	0,10	0,50	0,50	0,50	0,013	0,411	0
14	0,50	0,10	0,10	0,50	0,50	0,350	0	0
15	0,10	0,10	0,10	0,50	0,50	0,013	0	0,411
16	0,50	0,10	0,10	0,10	0,50	0,350	0	0
17	0,10	0,10	0,10	0,10	0,50	0,013	0	-0,411
18	0,50	0,10	0,10	0,10	0,10	0,350	0	0
19	0,10	0,10	0,10	0,10	0,10	0,013	-0,411	0

*The absence of obstacles determines the use of the average value of 0,5 on the universe [0, 1].

Table 4 – Obstacle maneuvering scenarios for quadcopter autopiloting

№	Inputs					Outputs		
	x_1	x_2	x_3	x_4	x_5	y_1	y_2	y_3
1	0,433	0,291	0,628	0,076	0,961	0,350	0,055	-0,190
2	0,611	0,824	0,460	0,862	0,855	0,380	-0,194	0,000
3	0,374	0,622	0,139	0,750	0,702	0,337	-0,154	0,215
4	0,712	0,738	0,568	0,438	0,469	0,854	0,000	0,000
5	0,922	0,535	0,320	0,107	0,102	0,855	0,000	0,000
6	0,396	0,283	0,318	0,925	0,001	0,349	0,000	0,204
7	0,026	0,563	0,311	0,177	0,959	0,013	-0,001	-0,410
8	0,627	0,457	0,127	0,826	0,117	0,467	0,000	0,171
9	0,570	0,359	0,197	0,404	0,663	0,351	0,000	-0,169
10	0,296	0,631	0,224	0,938	0,827	0,019	-0,097	0,384
11	0,426	0,026	0,464	0,954	0,842	0,350	0,000	0,200
12	0,602	0,563	0,781	0,415	0,181	0,363	0,197	0,000
13	0,510	0,253	0,232	0,894	0,429	0,350	0,000	0,200
14	0,333	0,292	0,856	0,275	0,762	0,131	0,386	0,000
15	0,645	0,632	0,452	0,743	0,643	0,658	-0,103	0,103
16	0,524	0,833	0,781	0,985	0,546	0,350	-0,200	0,000
17	0,152	0,557	0,453	0,289	1,000	0,013	-0,001	-0,410
18	0,709	0,310	0,242	0,138	0,410	0,853	0,000	0,000
19	0,389	0,113	0,828	0,625	0,121	0,347	0,209	0,000
20	0,133	0,292	0,972	0,499	0,693	0,013	0,411	0,000
21	0,534	0,148	0,532	0,721	0,969	0,350	0,000	0,200

Continuation of Table 4

22	0,492	0,921	0,376	0,131	0,585	0,350	-0,200	0,000
23	0,283	0,965	0,307	0,287	0,305	0,015	-0,410	0,000
24	0,178	0,407	0,298	0,385	0,427	0,016	-0,394	0,000
25	0,004	0,521	0,704	0,257	0,895	0,013	0,387	-0,088
26	0,557	0,562	0,291	0,562	0,910	0,350	-0,001	-0,200
27	0,296	0,592	0,922	0,910	0,104	0,019	0,400	0,000
28	0,189	0,123	0,004	0,101	0,411	0,015	-0,010	-0,399
29	0,410	0,491	0,864	0,569	0,522	0,350	0,201	0,000
30	0,605	0,961	0,766	0,756	0,306	0,367	-0,196	0,000
31	0,702	0,644	0,219	0,000	0,740	0,851	-0,001	-0,001
32	0,433	0,629	0,741	0,405	0,838	0,350	0,144	0,000
33	0,465	0,793	0,395	0,745	0,086	0,350	-0,200	0,000
34	0,065	0,004	0,789	0,553	0,318	0,013	0,411	0,000
35	0,026	0,245	0,480	0,651	0,787	0,014	0,000	-0,195

4. Neural network control of the flight path of a quadcopter

Table 4 presents various scenarios for the behavior of a quadcopter in the presence of obstacles in the five sectors of view during the process of overland monitoring. At the same time, the flight trajectory is formed based on the operational five-criteria assessment of the presence (or absence) of obstacles in all quadcopter viewing sectors, where the terms of input linguistic variables x_k $(k = 1 \div 5)$ act as qualitative evaluation criteria. An analytical approach to such assessment makes it possible to compare alternative quadcopter flight routes using the desired vector (y_1, y_2, y_3), reflecting the relative influence of factors x_k in the form of the mapping F: $R^5 \to R^3$. Therefore, it is advisable to represent the working model for the formation of the flight path of the quadcopter in the form of a "black box", where inputs and outputs are initially determined by expert judgements that constitute a heuristic knowledge base (an external representation of the given problem) in the form of information fragments $d_1 \div d_{19}$. Thus, to build an analytical model for the formation of the quadcopter flight trajectory, it is advisable to use a multilayer neural network, which, as known, is an effective tool for knowledge compilation relative to the behavior of a quadcopter under overland monitoring.

In the problem under consideration, "external knowledge" about 35 possible scenarios for the formation of the quadcopter's flight path is presented in the form of the following information model (see Table 4):

$$\{(x_{1j}, x_{2j}, x_{3j}, x_{4j}, x_{5j}) \to (y_{1j}, y_{2j}, y_{3j})\}_{j=1}^{35}.$$

Then the mapping F can be approximated by the three-layer feedforward neural network (FNN) (see Fig. 5), which induces a signal for each of the r-th outputs $(r = 1 \div 3)$ in the following form:

$$z_{rj} = \sum_{k=1}^{m} c_{rk} \varphi[\sum_{i=1}^{5} w_{ki} x_{ij} - \theta_k] \;\; (j = 1 \div 35),$$

where m is the number of nonlinear neurons in the "hidden" layer, selected by the user during the simulation by rule-of-thumb method; w_{ki} і $c_{rk} (k = 1 \div m)$ are the weights of input and output synaptic connection, respectively; θ_k is the threshold (bias) of the k-th nonlinear neuron from the "hidden" layer; $\varphi(\cdot)$ is the activation function of the non-linear neuron from the "hidden" layer, for example, sigmoidal type $\varphi(t) = 1/(1+e^{-t})$.

The neural network, which compiles the heuristic knowledge relative to the flight routing of the quadcopter, processes input vectors with components according to the number of criteria for assessing the "proximity" of obstacles in all sectors of the view, which are represented as numbers from the segment [0; 1]. As a result of the adjustment of the parameters (weights of synaptic connections and thresholds), the neural network can be approximate the imaginary mapping F: $R^5 \to R^3$, presented in tabular form (see Table 4). For each case, at its output, the neural network must generate command control of the quadcopter flight path in the form of the three-component vector (y_1, y_2, y_3) =(Airspeed, Yaw, Pitch). In particular, the neural network must respond with the output vector (0,350; 0,000; 0,200) at the input vector position (0,510; 0,253; 0,232; 0,894; 0,429)) (see scenario 13 from Table 4). As shown in Fig. 6, the network has one hidden layer, which consists of 5 nonlinear neurons with log-sigmoid activation functions, the range of which allows to realize: airspeed within the interval [0; 1], *yaw* within the interval [-0,5; 0,5], pitch within the interval [-0,5; 0,5].

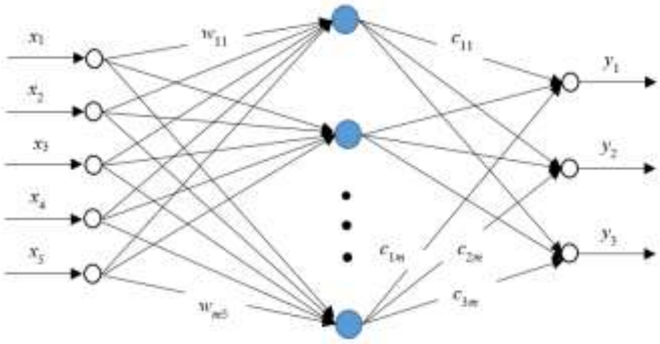

Figure 5 – The three-layer feedforward neural network

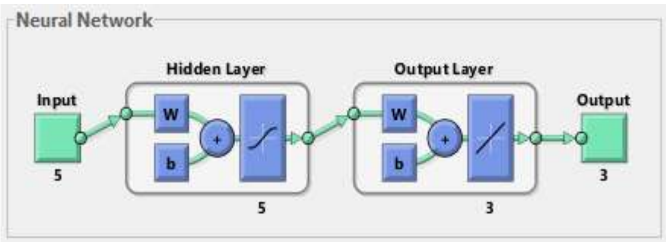

Figure 6 – Three-layer feedforward neural network in MATLAB notation

After training, testing, and validation of the neural network (see Fig. 7), the corresponding products are formed (results of pairs of the "input-output" type) and summarized in Table 5.

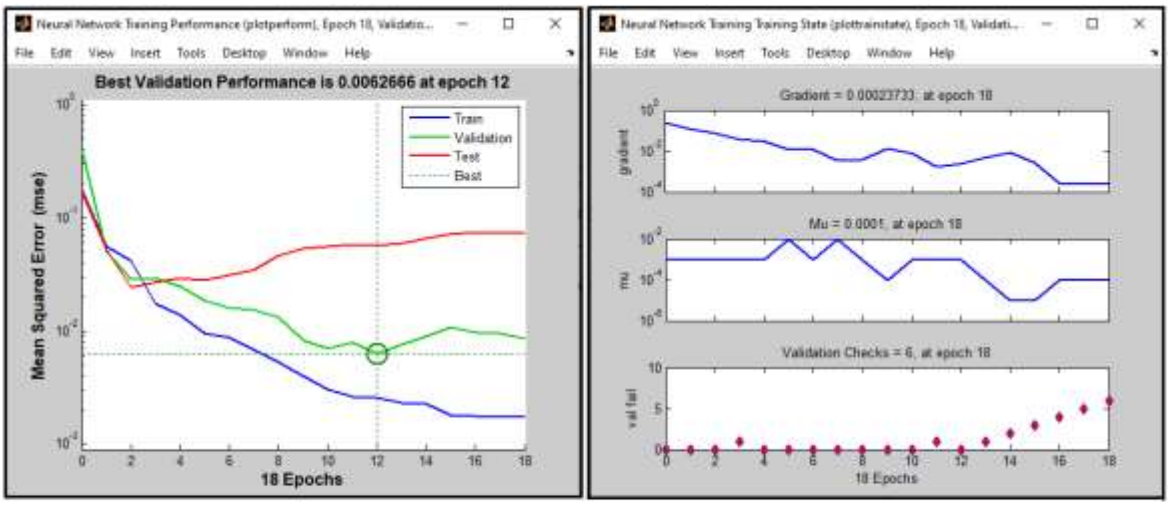

Figure 7 – Results of training, testing, and validation of the neural network in MATLAB notation

Table 5 – Control signals are generated by FNN

Scenario	Inputs					Outputs:					
						using FIS			using FNN		
	x_1	x_2	x_3	x_4	x_5	y_1	y_2	y_3	z_1	z_2	z_3
1	0,433	0,291	0,628	0,076	0,961	0,350	0,055	-0,190	0,3805	0,0638	-0,2155
2	0,611	0,824	0,460	0,862	0,855	0,380	-0,194	0,000	0,3720	-0,2092	0,0435
3	0,374	0,622	0,139	0,750	0,702	0,337	-0,154	0,215	0,1603	-0,1010	0,1908
4	0,712	0,738	0,568	0,438	0,469	0,854	0,000	0,000	0,8450	-0,0334	0,0307
5	0,922	0,535	0,320	0,107	0,102	0,855	0,000	0,000	0,8925	0,0019	-0,0344
6	0,396	0,283	0,318	0,925	0,001	0,349	0,000	0,204	0,3445	-0,0126	0,1790
7	0,026	0,563	0,311	0,177	0,959	0,013	-0,001	-0,410	0,0161	-0,0592	-0,3609
8	0,627	0,457	0,127	0,826	0,117	0,467	0,000	0,171	0,4858	0,0041	0,2113
9	0,570	0,359	0,197	0,404	0,663	0,351	0,000	-0,169	0,3067	0,0290	-0,1231
10	0,296	0,631	0,224	0,938	0,827	0,019	-0,097	0,384	0,1858	-0,1288	0,3603
11	0,426	0,026	0,464	0,954	0,842	0,350	0,000	0,200	0,3424	0,0212	0,2167
12	0,602	0,563	0,781	0,415	0,181	0,363	0,197	0,000	0,8288	-0,1886	0,3207
13	0,510	0,253	0,232	0,894	0,429	0,350	0,000	0,200	0,3598	-0,0156	0,1699
14	0,333	0,292	0,856	0,275	0,762	0,131	0,386	0,000	0,1945	0,3646	-0,0119
15	0,645	0,632	0,452	0,743	0,643	0,658	-0,103	0,103	0,6319	-0,0920	0,2408
16	0,524	0,833	0,781	0,985	0,546	0,350	-0,200	0,000	0,3832	-0,0907	0,1373
17	0,152	0,557	0,453	0,289	1,000	0,013	-0,001	-0,410	0,0147	0,0025	-0,3394
18	0,709	0,310	0,242	0,138	0,410	0,853	0,000	0,000	0,8375	0,0191	-0,0102
19	0,389	0,113	0,828	0,625	0,121	0,347	0,209	0,000	0,2847	0,2029	0,0528
20	0,133	0,292	0,972	0,499	0,693	0,013	0,411	0,000	0,0004	0,4498	-0,0840
21	0,534	0,148	0,532	0,721	0,969	0,350	0,000	0,200	0,3557	0,0093	0,1589
22	0,492	0,921	0,376	0,131	0,585	0,350	-0,200	0,000	0,4234	-0,0783	-0,1872
23	0,283	0,965	0,307	0,287	0,305	0,015	-0,410	0,000	0,0327	-0,3269	-0,1404
24	0,178	0,407	0,298	0,385	0,427	0,016	-0,394	0,000	0,0261	-0,3941	-0,0363
25	0,004	0,521	0,704	0,257	0,895	0,013	0,387	-0,088	-0,0061	0,4013	-0,0879
26	0,557	0,562	0,291	0,562	0,910	0,350	-0,001	-0,200	0,1962	-0,0842	-0,2622
27	0,296	0,592	0,922	0,910	0,104	0,019	0,400	0,000	0,0552	0,3839	-0,0457
28	0,189	0,123	0,004	0,101	0,411	0,015	-0,010	-0,399	0,0470	-0,1082	-0,3173
29	0,410	0,491	0,864	0,569	0,522	0,350	0,201	0,000	0,3144	0,2509	0,0694
30	0,605	0,961	0,766	0,756	0,306	0,367	-0,196	0,000	0,7688	-0,1998	0,3088
31	0,702	0,644	0,219	0,000	0,740	0,851	-0,001	-0,001	0,8111	-0,0211	-0,0081
32	0,433	0,629	0,741	0,405	0,838	0,350	0,144	0,000	0,3401	0,1108	-0,0560
33	0,465	0,793	0,395	0,745	0,086	0,350	-0,200	0,000	0,1829	-0,2772	0,1751
34	0,065	0,004	0,789	0,553	0,318	0,013	0,411	0,000	0,0047	0,4062	-0,0850
35	0,026	0,245	0,480	0,651	0,787	0,014	0,000	-0,195	0,0159	-0,0078	-0,0427

5. Conclusion

Complexity, multi coupling, nonlinearity, the presence of uncertain parameters of such technical objects as a quadcopter necessitates the search for ways to solve control problems alternative to the well-known classical P, PI, PID controllers. The result of the expert-empirical research carried out in this article is to get an idea of how to control quadcopters under overland monitoring of the area, the features of their practical application, advantages, and disadvantages. It seems to us that the improvement of results should be achieved using neural and neuro-fuzzy modeling methods. For example, any membership function given at key points obtained using Saaty's 9-point scale can be easily approximated using a three-layer feedforward neural network. Based on the scenarios presented in Table 4, it is possible to form a neuro-fuzzy controller capable of providing overland autopiloting of a quadcopter under different obstacles.

REFERENCES

1. Michael D.S. Simulation and control of a quadrotor unmanned aerial vehicle. URL: http://uknowledge.uky.edu/gradschool_theses/93.
2. Sharma A., Barve A. Controlling of quadrotor UAV using PID controller and fuzzy logic controller. *International Journal of Electrical, Electronics and Computer Engineering.* 2021. Vol. 1 (2). P. 38–41. DOI: http://doi.org/10.11591/ijece.v2i1.
3. DJI drones: types and capabilities of the brand's aircraft. URL: https://korabelov.info/2021/11/220373/drony-dji-vidy-i-vozmozhnosti-letatelnyh-apparatov-brenda/ (in Russian).
4. Neretina V.V., Nafikov I.R. Formation of the UAV trajectory during overland monitoring of forest eco-systems based on the apparatus of fuzzy logic. URL: https://apni.ru/article/845-formirovanie-traektorii-dvizheniya-bpla.
5. Saaty T.L., Vargas L.G. The Logic of Priorities. Applications of Business, Energy, Health and Transportation. Kluwer Nijhoff Publishing, Massachusetts, 1982. 278 p.

Стаття надійшла до редакції 22.07.2022