Takagi-Sugeno Fuzzy Logic for Reaction Pendulum

January, 2025

Author:
Pawel Wiktor

Email:
pawelwiktor.kontakt@gmail.com

GitHub:
https://https://github.com/pawelwiktor-github

Table of Contents

1 Creating Fuzzy Logic Takagi-Sugeno Controller (MISO) for Pendulum Control Signal

In this task, the objective was to design a Takagi-Sugeno fuzzy logic controller, with the target of stabilizing the reaction pendulum in a fixed position. The whole process was carried out in the Matlab-Simulink environment, using a simulation model of the reaction pendulum. The side objective was to surpass LQR controller in the execution of effective stabilisation.

1.1 Theoretical Introduction

Controllers based on fuzzy logic are widely used in control systems, especially where there are nonlinearities and uncertainties. Prominent among them is the Takagi-Sugeno (TSK) type controller, which is characterized by a specific structure of rules and output functions.

The TSK regulator differs from previously created Mamdani primarily in the form of the consequent part of the rules. In TSK, the rules take the form:

"If x is A and y is B, then z = f(x, y)"

where $f(x, y)$ is an output function, either linear or constant. This approach allows more precise modeling of nonlinear systems through local linear approximations.

Depending on the complexity of the system being modeled and the control accuracy requirements, different types of output functions are used in the TSK controller:

- **Constant:** The simplest form, where the output is a fixed value. Used in systems with low dynamics or when faster reaction is required.
- **Linear:** Linear functions that depend on inputs, allowing for a better representation of system dynamics. For example, a function may be of the form $z = ax + by + c$, where a, b and c are parameters to be determined.

Advantages of the TSk controller:

- **Better accuracy:** By using input-dependent output functions, the TSK controller can more accurately model complex nonlinear systems.
- **Ease of integration:** Rules with linear functions can be easily integrated with classical control methods such as PID, making it possible to create hybrid control systems.
- **Computational efficiency:** Compared to Mamdani controllers, TSK often requires fewer rules to achieve comparable accuracy, resulting in a lower computational load.

1.2 Creating Takagi-Sugeno fuzzy logic controller

In order to properly design a controller for any system, one must first analyze the dynamics of its operation. The first important thing for this purpose was to know the ranges over which the controller's input signals work: pendulum angle, pendulum velocity and motor velocity. They were determined on the basis of experiments performed on the reaction pendulum related to:

- the maximum possible angle of deviation from zero, for which the LQR controller managed to stabilize the pendulum
- the maximum velocity that the pendulum reached during its free fall from the upper position

and by using the datasheets - the maximum speed of the motor. Determined ranges are presented in the table 1.

Table 1. Fuzzy logic controller input signal ranges

Signal	Range
Pendulum Angle [°]	$-30 - 30$
Pendulum Velocity $[rad/s]$	$-4.2 - 4.2$
Motor Velocity $[rad/s]$	$-450 - 450$

The second important step is to analyze how the *Reaction Pendulum* model works. The analysis process was based on *LQR Benchmark Controller* which reacted to the 30° swing of the Pendulum. The swing in oposite way showed exactly opposite behaviour (same ranges, different direction).

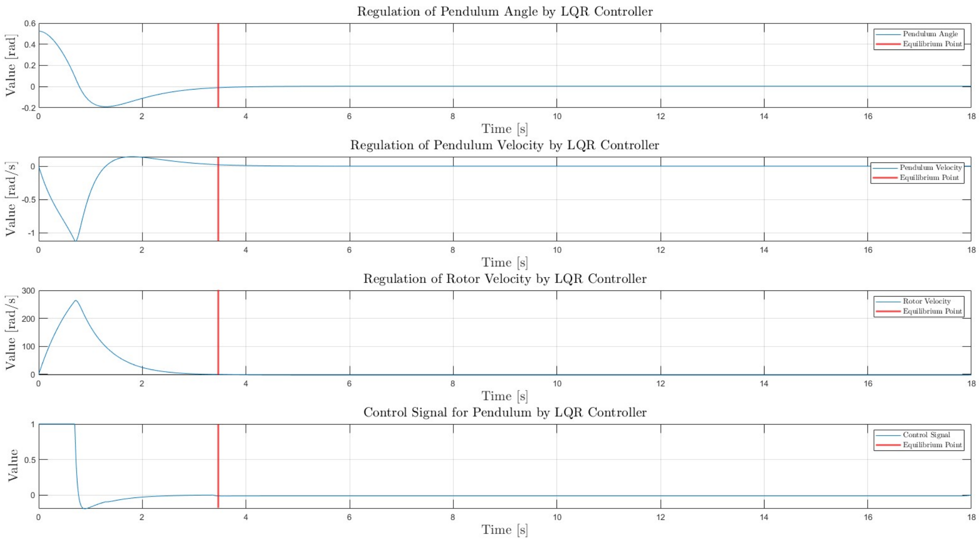

Fig. 1. *LQR* regulation of *Pendulum* after 30° swing

The first thing observed is that when the Pendulum is swinged, to bring it back to equilibrium point, the control signal needs to spin the rotor disc in the opposite direction.

In this case, when the Pendulum Angle is equal to 30° and it is starting to go to the equilibrium point, the Control Signal is maximum and the motor is starting to spin in the opposite direction. Accordingly, the Pendulum arm is heading to the left. During the process, the control signal is lower in value.

When Pendulum Angle slightly shifts to the left side, the rotor is starting to brake and Control Signal is coming back to zero.

The equilibrium point is reached at 3.472 [s].

With that knowledge, it is possible to create membership functions to influence the behavior of the model for specific cases.

1.3 Membership Functions

Membership functions were created based on previous knowledge of the model. It was realized that the least number of membership functions for the Fuzzy Logic Controller to work properly is 3 (for inputs and output). The triangular membership function were chosen because of its simplicity.

Specifications are shown in the tables 2, 3, 4, 5.

Table 2. Pendulum Angle Membership Functions

Name	Type	Values
Pendulum Angle is Left	Triangular	$[-3.14159, -0.5, 0]$
Pendulum Angle is Center	Triangular	$[-0.08, 0, 0.08]$
Pendulum Angle is Right	Triangular	$[0, 0.5, 3.14159]$

Table 3. Pendulum Velocity Membership Functions

Name	Type	Values
Pendulum Velocity is Left	Triangular	$[-4.2, -2.1, 0]$
Pendulum Velocity is NoMovement	Triangular	$[-0.12, 0, 0.12]$
Pendulum Velocity is Right	Triangular	$[0, 2.1, 4.2]$

Table 4. Rotor Velocity Membership Functions

Name	Type	Values
Rotor Velocity is Negative	Triangular	$[-450, -260, 0]$
Rotor Velocity is Zero	Triangular	$[-30, 0, 30]$
Rotor Velocity is Positive	Triangular	$[0, 260, 450]$

Table 5. Control Signal Membership Functions

Name	Type	Values
Control Signal is Negative	Constant	-0.95
Control Signal is Zero	Constant	0
Control Signal is Positive	Constant	0.95

1.4 Rules

Rules specify the behavior of the whole system based on membership functions. That means the controller should react this way if this parameter has this value. It is necessary to cover all the cases that are necessary for the control signal to be working properly. To be able to cover all cases, you need to visualize how system should work and have a priori knowledge of the model that you are working on (or have specific experiments that show types of behavior).

Created rules are shown in the table 6.

Table 6. Fuzzy Rules for Pendulum Control

Rule	Description	Weight
1	If PendulumAngle is Left and PendulumVelocity is Left and RotorVelocity is Negative then Control is Negative	1
2	If PendulumAngle is Center and PendulumVelocity is Left and RotorVelocity is Negative then Control is Negative	1
3	If PendulumAngle is Right and PendulumVelocity is Left and RotorVelocity is Negative then Control is Positive	1
4	If PendulumAngle is Left and PendulumVelocity is Zero and RotorVelocity is Negative then Control is Negative	1
5	If PendulumAngle is Center and PendulumVelocity is Zero and RotorVelocity is Negative then Control is Negative	1
6	If PendulumAngle is Right and PendulumVelocity is Zero and RotorVelocity is Negative then Control is Positive	1
7	If PendulumAngle is Left and PendulumVelocity is Right and RotorVelocity is Negative then Control is Negative	1
8	If PendulumAngle is Center and PendulumVelocity is Right and RotorVelocity is Negative then Control is Positive	1
9	If PendulumAngle is Right and PendulumVelocity is Right and RotorVelocity is Negative then Control is Positive	1
10	If PendulumAngle is Left and PendulumVelocity is Left and RotorVelocity is Zero then Control is Negative	1
11	If PendulumAngle is Center and PendulumVelocity is Left and RotorVelocity is Zero then Control is Negative	1
12	If PendulumAngle is Right and PendulumVelocity is Left and RotorVelocity is Zero then Control is Positive	1
13	If PendulumAngle is Left and PendulumVelocity is Zero and RotorVelocity is Zero then Control is Negative	1
14	If PendulumAngle is Center and PendulumVelocity is Zero and RotorVelocity is Zero then Control is Zero	1
15	If PendulumAngle is Right and PendulumVelocity is Zero and RotorVelocity is Zero then Control is Positive	1
16	If PendulumAngle is Left and PendulumVelocity is Right and RotorVelocity is Zero then Control is Negative	1
17	If PendulumAngle is Center and PendulumVelocity is Right and RotorVelocity is Zero then Control is Positive	1
18	If PendulumAngle is Right and PendulumVelocity is Right and RotorVelocity is Zero then Control is Positive	1
19	If PendulumAngle is Left and PendulumVelocity is Left and RotorVelocity is Positive then Control is Negative	1
20	If PendulumAngle is Center and PendulumVelocity is Left and RotorVelocity is Positive then Control is Negative	1
21	If PendulumAngle is Right and PendulumVelocity is Left and RotorVelocity is Positive then Control is Positive	1
22	If PendulumAngle is Left and PendulumVelocity is Zero and RotorVelocity is Positive then Control is Negative	1
23	If PendulumAngle is Center and PendulumVelocity is Zero and RotorVelocity is Positive then Control is Positive	1
24	If PendulumAngle is Right and PendulumVelocity is Zero and RotorVelocity is Positive then Control is Positive	1
25	If PendulumAngle is Left and PendulumVelocity is Right and RotorVelocity is Positive then Control is Negative	1
26	If PendulumAngle is Center and PendulumVelocity is Right and RotorVelocity is Positive then Control is Positive	1
27	If PendulumAngle is Right and PendulumVelocity is Right and RotorVelocity is Positive then Control is Positive	1

1.5 Reviewing FIS structure

Fuzzy Logic structure is shown in figure 2.

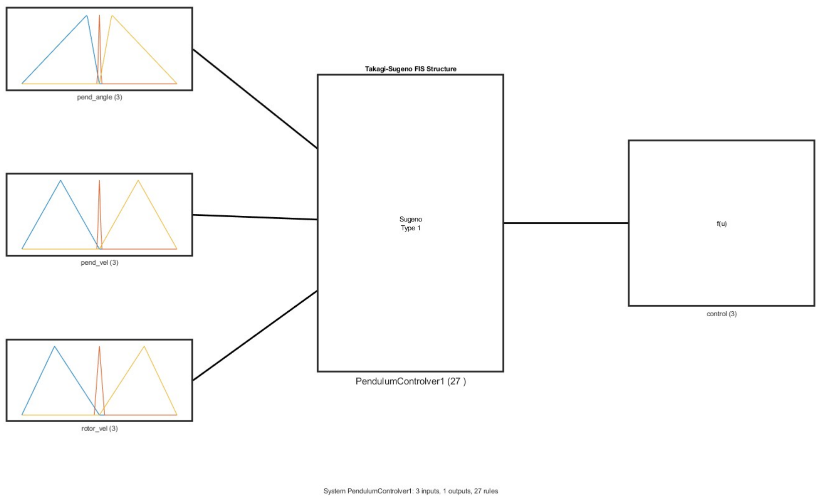

Fig. 2. *Takagi-Sugeno Fuzzy Controller structure*

Matlab environment enables the user to visualize graphically the membership functions created using the *plotmf* function. Those graphical illustrations are shown in figure 3.

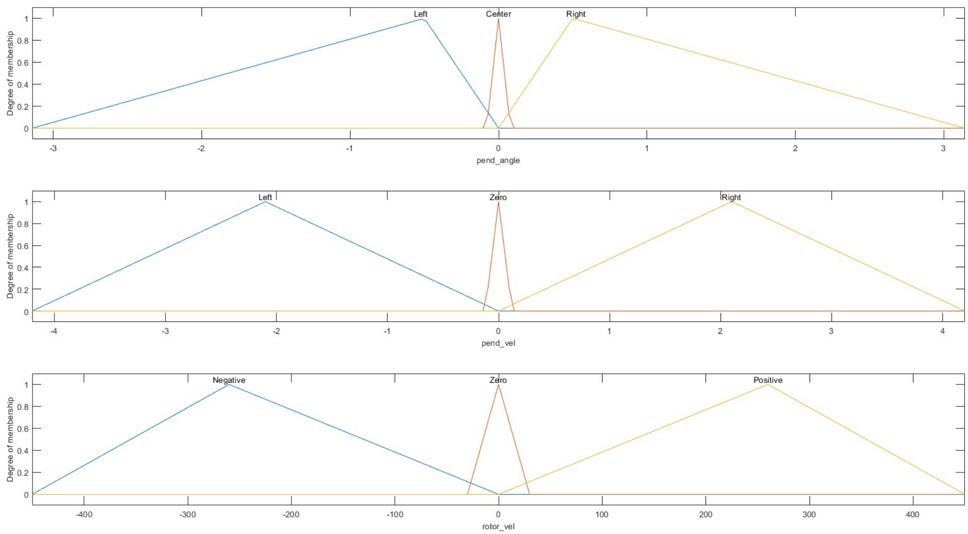

Fig. 3. Membership Function visualization

After rules creation there is a possibility to visualize control surface which shows how Control Signal is reacting for specific values of parameters (followed by the rules). Because we have 4-dimensional parameters, Matlab does not allow user to create that type of graph. That is why there are 3 figures that contain all possibilities of input parameters compared to output one. It is shown in figure 4.

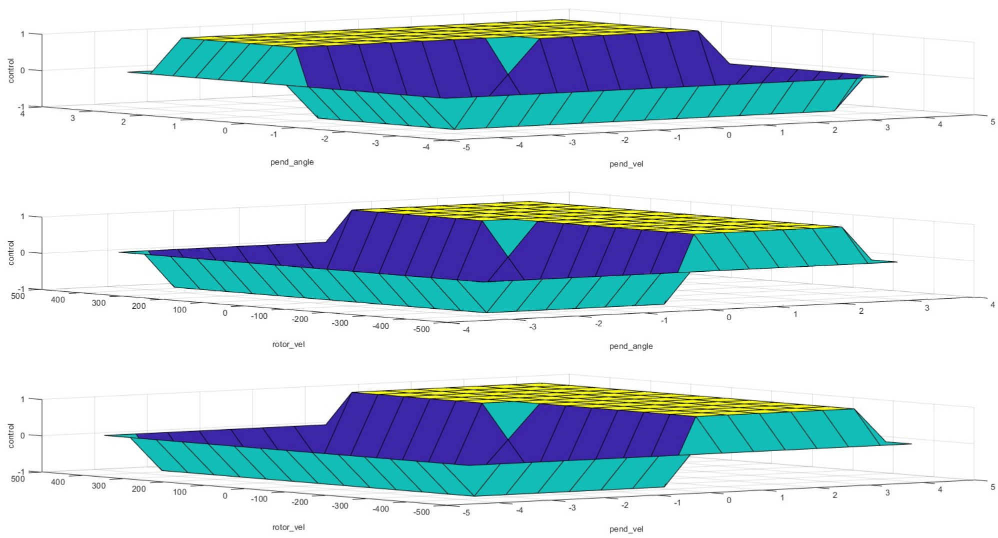

Fig. 4. Control surface visualization

1.6 Conlcusions

Figure 5 represents comparison to LQR (under the same conditions).

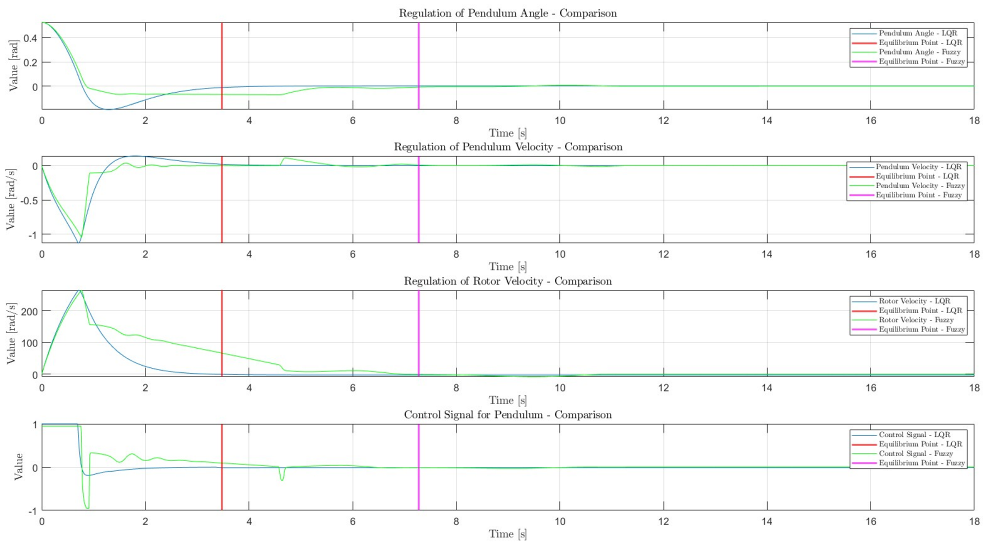

Fig. 5. Comparison of controllers regulation of Pendulum after 30° swing

The first visible issue with Fuzzy is that when the angle reaches around the center position, there are some overregulation in Pendulum Angle, Pendulum Velocity, Rotor Velocity, and Control Signal. In my opinion, it is because of insufficient regulation, i.e. poorly matched ranges of membership functions. It can cause engine overheating and loss of efficiency of the system.

Comparison of controllers shows that apart from overregulations Fuzzy stabilized Pendulum 3.801 [s] later then original LQR. The problem can occur through misalignment of membership functions.

Additionally by reviewing theory of fuzzy logic reminded that because of it's accuracy of control selection by construction of *if-then* rules, the Fuzzy should work faster and smoother than LQR. It was discovered that the issue is caused by too few membership functions for the control with 27 rules (there should be 9 membership functions for the control in this scenario). It is caused due to used Fuzzy Creation Parameters. In this project it was used respectively: And method(min), Implication method(prod), Aggregation method(sum), Defuzzification method(wtaver). Description can be found below:

- And Method == min -> Uses the minimum value of the membership degrees
- Implication Method == prod -> Uses the product of the antecedent truth value and the membership function of the consequent
- Aggregation Method == sum -> Adds the outputs of the rules together
- Defuzzification Method == wtaver -> The defuzzified output is calculated as a weighted average of the rule outputs

By using this approach, the output is calcualted by degree of membership and rule weight multiplication and then there is a sum of applied rules. In the end, by defuzzification method, output value is a weighted average of mentioned above mathematical operations. Based on this observation, it was found that far fewer rules could be designed i.e. 3 for each input.

1.7 Fuzzy Logic Takagi-Sugeno Controller correction

1.7.1 Membership Functions correction

Apart from changing Fuzzy Creation Parameters, Control Signal Membership Functions (table 7) were completely changed.

Table 7. Control Signal corrected Membership Functions

Name	**Type**	**Values**
Control Signal is Negative	Constant	-6.7333
Control Signal is Zero	Constant	0
Control Signal is Positive	Constant	6.7333

1.7.2 Rules correction

Table 8. Corrected Fuzzy Rules for Pendulum Control

Rule	**Description**	**Weight**
1	If PendulumAngle is Left then Control is Positive	0.0714
2	If PendulumAngle is Center then Control is Zero	0.0714
3	If PendulumAngle is Right then Control is Negative	0.0714
4	If PendulumVelocity is Left then Control is Positive	0.159
5	If PendulumVelocity is Zero then Control is Zero	0.159
6	If PendulumVelocity is Right then Control is Negative	0.159
7	If RotorVelocity is Negative then Control is Positive	0.0995
8	If RotorVelocity is Zero then Control is Zero	0.0995
9	If RotorVelocity is Positive then Control is Negative	0.0995

Rule wages were changed in order to adapt Control Signal multiplication for specified parameter in addition to reducing the number of rules.

1.8 Reviewing FIS structure after correction

Fuzzy Logic structure is shown in figure 6.

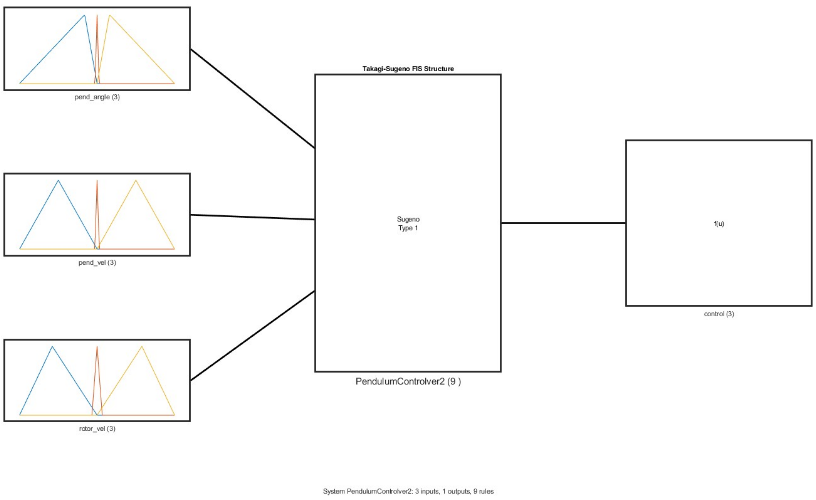

Fig. 6. *Takagi-Sugeno Fuzzy Controller structure*

After correction results of membership functions and control surface are presented respectively on figure 7 and 8.

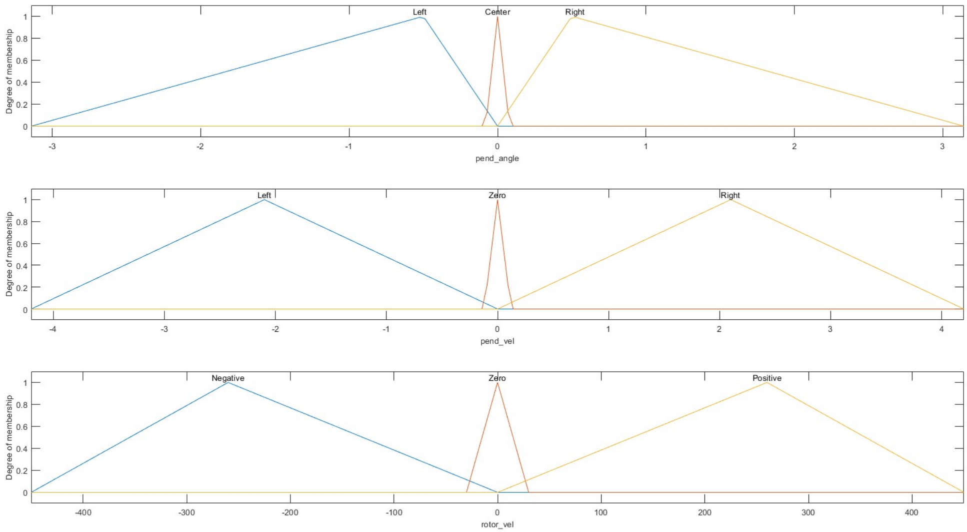

Fig. 7. Membership Function visualization

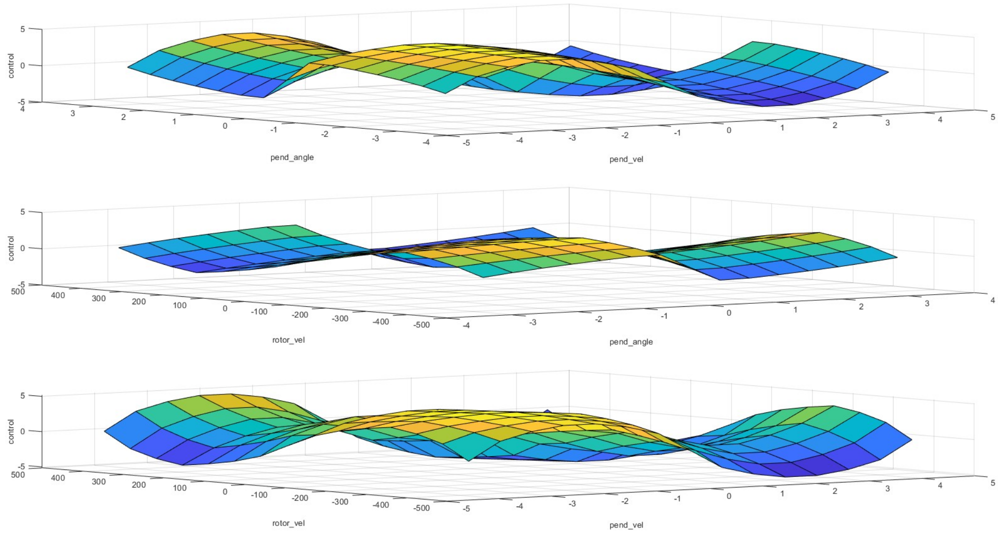

Fig. 8. Control surface visualization

1.8.1 Fuzzy Logic Takagi-Sugeno Controller correction results

After the correction, the performance of Fuzzy was much better. Trasitions between states are smoother, and stabilization is achieved faster.

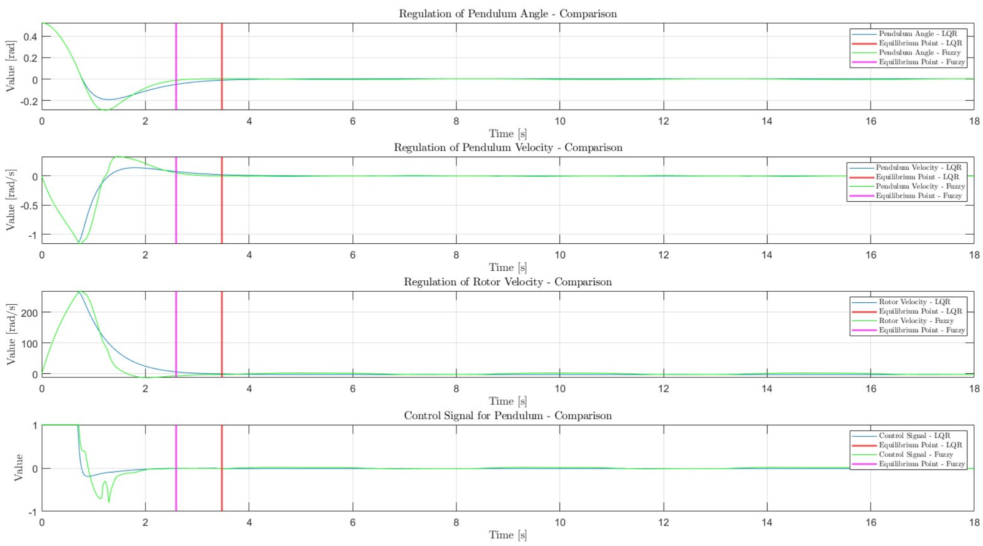

Fig. 9. Comparison of controllers regulation of Pendulum after 30° swing

Fuzzy performs stabilization at 2.597 [s], which is 0.88 [s] faster than LQR.

1.9 Quality indicators

In order to check the robustness of the controller, two integral quality indices were calculated each for the different experiments carried out and the stabilization time of the pendulum was manually read and analyzed:

- Integral from the square of the control deviation (J_1)

- Integral of the absolute value of the control multiplied by time (J_2)
- Stabilization time (t_s)

Equations for quality indicators:

$$J_1 = \int_0^T u^2(t)\, dt \tag{1}$$

$$J_2 = \int_0^T |u(t)| \cdot t\, dt \tag{2}$$

where:

J_1	– Integral from the square of the control deviation
J_2	– Integral of the absolute value of the control multiplied by time
$u(t)$	– control value in time
t	– simulation time

Description of experiments:

1) Constant disturbance in the deflection of the pendulum from equilibrium by 0.2 rad every 10 seconds
2) Constant disturbance in the deflection of the pendulum from equilibrium by 0.2 rad every 10 seconds + increased distance from the pivot point
3) Constant disturbance in the deflection of the pendulum from equilibrium by 0.2 rad every 10 seconds + constant motor disc rotation speed of 100 rad/s

Results for each experiment (per one swing):

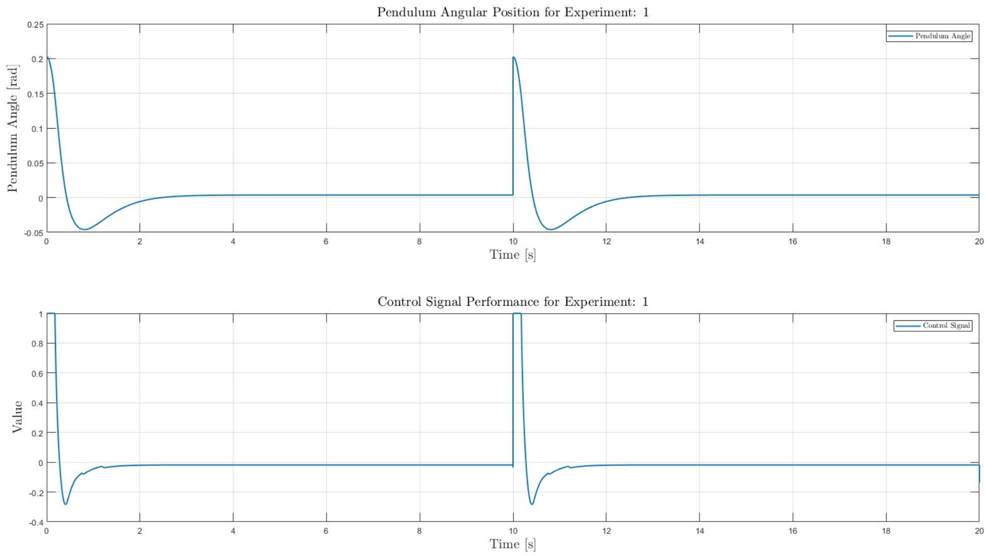

Fig. 10. Results for first experiment

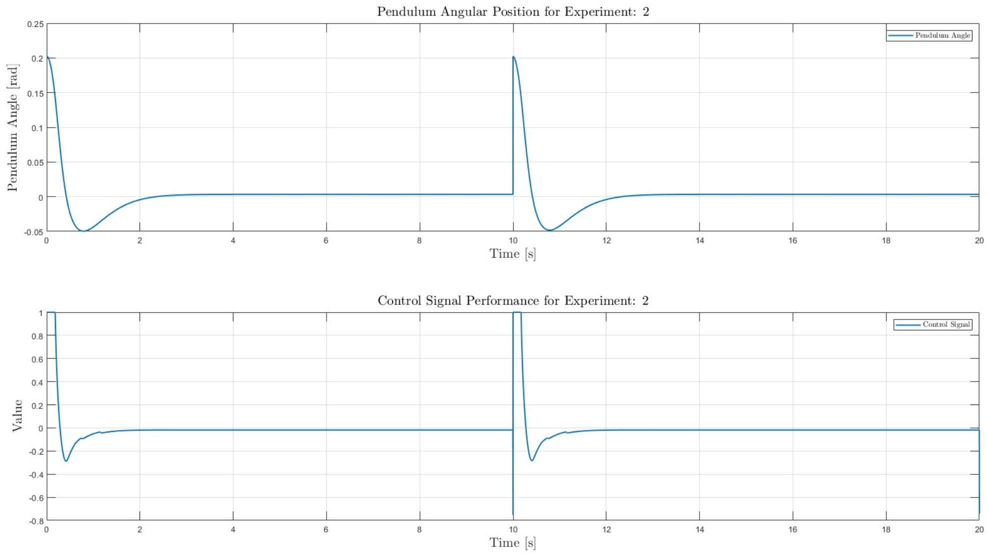

Fig. 11. Results for second experiment

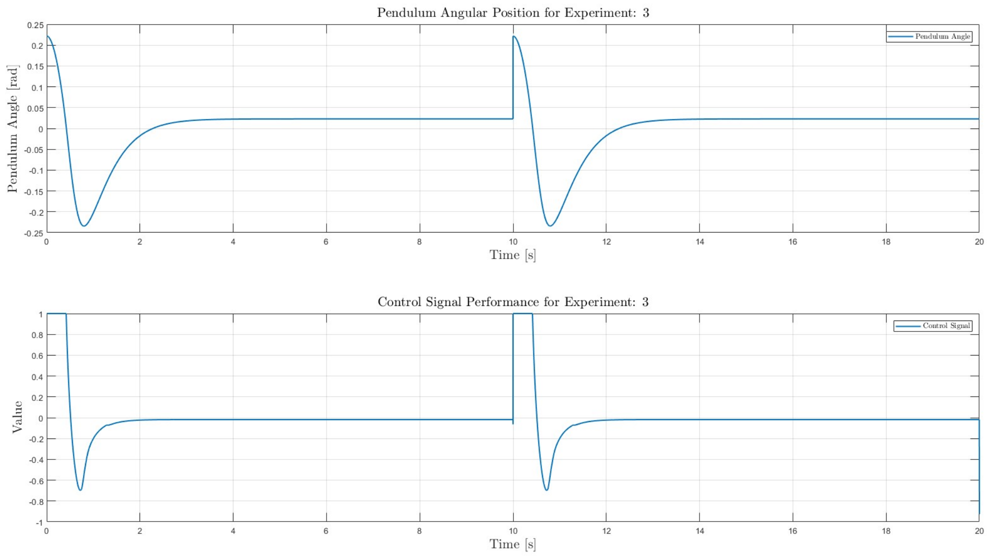

Fig. 12. Results for third experiment

Quality indicators were calculated and shown in the table 9.

Table 9. Quality indicators calculation results

Experiment indicator	J_1	J_2	t_s
First experiment	0.4314	2.0162	1.9428
Second experiment	0.4672	2.1528	1.9843
Third experiment	1.1606	3.9015	2.0997

1.9.1 Conclusions

Quality indicators validated the robustness and efficiency of the controller under varying disturbances. The stabilization time t_s revealed that the controller performed the best in the first experiment. The increased distance from the pivot point slightly influenced the stabilization performance within an acceptable range. The values of J_1 and J_2 indicated minimal control deviation and stable control application, demonstrating the effectiveness of the Takagi-Sugeno controller in managing disturbances for the first and second experiments.

The results of the third experiment have shown that Fuzzy is trying to stabilize the Pendulum arm in the top position, but with a constant motor disc rotation speed set to 100 $[\frac{rad}{s}]$ it is impossible for the system to stabilize in 0. It is caused by an insufficient control signal shown by the J_1 indicator (which exceeded the [-1 1] range). Instead, Fuzzy stabilize Pendulum at around 11[°] (figure 12).

1.10 Summary

Objective

The study aimed to design a Takagi-Sugeno fuzzy logic controller for stabilizing a reaction pendulum, using MATLAB-Simulink. The controller was benchmarked against a Linear Quadratic Regulator (LQR) to assess its performance and effectiveness.

Key Features of Takagi-Sugeno Controller

- **Theoretical Basis:** Utilizes rules of the form "If x is A and y is B, then $z = f(x, y)$," where the output functions are either linear or constant.
- **Advantages:**
  - High modeling precision for nonlinear systems.
  - Efficient integration with classical control methods like PID.
  - Requires fewer rules, resulting in lower computational load.

Controller Design

- **Input Parameters:** Pendulum angle, pendulum velocity, and motor velocity, with their respective ranges determined experimentally.
- **Membership Functions:** Initially three triangular membership functions per parameter.
- **Rules:** 27 rules defined based on the model's behavior.
- **Fuzzy Inference System (FIS):** Created using MATLAB to visualize membership functions and control surface.

Results

The controller achieved stabilization but with overregulation issues near the equilibrium point, resulting in slower stabilization compared to LQR.

Corrections and Improvements

1. **Membership Functions:** Modified to increase precision and address overregulation.
2. **Rules:** Reduced from 27 to 9, improving efficiency and clarity.
3. **Results After Correction:**
   - Faster stabilization time (2.597s compared to LQR's 3.477s).
   - Smoother state transitions with improved control performance.

Quality Indicators

Performance was evaluated using:

- Integral of squared control deviation (J_1).
- Integral of absolute control multiplied by time (J_2).
- Stabilization time (t_s).

Findings:

- The controller performed robustly under varying disturbances in two experiments but struggled in the third experiment due to a high constant motor speed.

Conclusions

The corrected Takagi-Sugeno controller demonstrated superior performance compared to LQR, achieving faster and smoother stabilization. However, extreme scenarios revealed limitations in the control signal's range, highlighting areas for further refinement. Overall, the study validated the effectiveness and adaptability of fuzzy logic in nonlinear system control.