Vietnam Journal of Science and Technology ***61*** *(4) (2023) 531-569*

doi:10.15625/2525-2518/18069

Application of fuzzy logic in the robot control for mechanical processing

Phan Bui Khoi

Hanoi University of Science and Technology, No. 1, Dai Co Viet, Hai Ba Trung District, Ha Noi, Viet Nam

Email: *khoi.phanbui@hust.edu.vn*

Received: 8 February 2023; Accepted for publication: 1 June 2023

Abstract. Robot application in mechanical machining is growing daily because it has many advantages over conventional machines, such as high flexibility, large working space, and high repeatability. Many degrees of freedom of motion give robots the ability to perform complex technological operations, but also because of that, methods of controlling robots based on dynamic models have difficulties. Applying fuzzy logic to robot control can partially or entirely exclude the calculation of the robot's dynamic model as well as overcome other uncertainties of the whole technological system. Fuzzy rules are an essential basis for performing operations defining the control quantities of the controller. The main tasks of applying fuzzy logic control include "Fuzzification" to determine fuzzy parameters in the form of fuzzy sets of input-output data, "Fuzzy Rules and Fuzzy Inference Mechanism" to perform fuzzy operations defining control quantities, and finally, "Defuzzification" to convert control quantities from linguistic values to physical values for controller operation. Belonging to the above contents, the core issues to ensure the quality and effectiveness of fuzzy controllers are reasonable determination: (1) physical value domain of input and output linguistic variables, (2) fuzzy rule base system and fuzzy inference mechanism, (3) algorithms for dealing with fuzzy relationships and defuzzification. There have been many types of research on applying fuzzy logic in control engineering in general, as well as robot control. However, the above core problems are still open problems and challenges. On the other hand, the percentage of fuzzy control research work for mechanical processing robots is still limited.

This article is based on published works on fuzzy control for robots in general and mechanical processing robots to analyze the applicability of fuzzy control for mechanical machining robots. The article provides detailed information on fuzzy controller design, determining input and output variables, and proportional mapping to determine the number of fuzzy sets and the corresponding type of membership function. The construction of a Fuzzy Rule base system and Fuzzy Inference Mechanism and Defuzzification is presented. Prospects for using methods to perfect and develop fuzzy control systems for mechanical machining robots are also presented. Collectively, the information in this article is intended to guide the implementation of fuzzy logic-based controller designs for application to machining robots, as well as general robot control.

Keywords: fuzzy logic, mechanical processing robot, robot control, fuzzy control, fuzzy law.

Classification numbers: 5.3.7, 5.3.8, 5.8.1

1. INTRODUCTION

Robots are widely applied in many fields of production and service life. Robotic mechanical processing has many advantages thanks to the dynamic multi-link structure for flexible manipulation, large working space, and high repeatability [1 - 11].

One of the top tasks when applying robots in general and in mechanical processing is to control the robot to perform operations that meet technical requirements. Suppose the dynamic model of the system is determined precisely. In that case, a robotic controller based on the system dynamics model, hereinafter referred to as a crisp controller, can be easily implemented and ensures the required accuracy.

As is known, due to the complex structure, the dynamic model of the robot is often complex to fully and accurately determine. With mechanical processing robots, due to manipulation over time, some dynamic parameters often change uncertainly, such as cutting force, friction, vibration, and impact, etc.

Many methods of building crisp controllers with correction components have been studied and applied to overcome difficulties in determining and calculating dynamic models, such as adaptive controllers, stable controls, adaptive slide control, optimal control, etc.

The application of fuzzy logic to control techniques has been widely studied and applied. The advantage of fuzzy logic-based control is simplicity and ease of implementation [12 - 27].

In human activity, to represent the world, communicate and process information, make decisions or perform tasks, etc., one can use language, the rules of logical reasoning, and also operate with sets of objects presented in a linguistic form, with linguistic semantic values, etc.

Fuzzy logic is an imitation of natural human reasoning applied in general practice and control engineering [16 - 22]. Accordingly, variables and parameters represent objects in natural language form, called linguistic variables, with linguistic semantic values. A rule system, also known as a rule base, and an inference mechanism built for mathematic operations on sets (fuzzy sets) of linguistic variables. The problem for a controller is how to make the robot, as well as the machines and devices, operate according to the rules of natural inference, like humans, from the incomplete information provided by the fuzzy sets.

There have been many studies, and as shown in most of the works, a controller is based on fuzzy logic, hereinafter referred to as the fuzzy controller, usually structured by three main blocks: Fuzzification, Fuzzy Composition Device, and Defuzzification. The two parts of the Fuzzification block include the Database and the Fuzzification to convert the input and output databases into fuzzy sets. The Fuzzy Composition Devices block consists of the Fuzzy Rule Base and the Fuzzy Inference Mechanism, integrated by algorithms to process databases in the form of fuzzy sets. The Defuzzification block converts control quantities from linguistic values to physical values that enable the system to function.

The operations of the controller blocks are performed based on mathematical models built for fuzzy algorithms.

The mathematical model of the fuzzy set and the fuzzy logic concept was presented by Lotfi Zadeh, starting in 1965, about the fuzzy set [12]. However, fuzzy logic has been studied before, since the 1920s, as logic with infinite values, especially by Łukasiewicz and Tarski [10,11].

The research and application of fuzzy set theory and fuzzy logic have achieved remarkable achievements. Fuzzy theory is applied in almost all fields of science and engineering, most commonly in control engineering. Therefore, fuzzy logic has been used to control virtually all types of robots from service to industry, and some works can be briefly mentioned below.

The works [12 - 56] present the application of fuzzy logic in control engineering and robot control in general. Some results on controlling arm-shaped robots perform technological manipulation in general and mechanical processing, as shown in [57 - 81]. The most popular research and application is the fuzzy control of mobile robots for robot positioning and navigation. In [82-92] are some of the many such works.

It can be seen that the studies and results on the use of fuzzy logic in the control of mechanical machining robots are relatively less as compared to the control of mobile robots. The reason is simple: mobile robots do not have high precision requirements such as those of mechanical processing robots, and the structure of mobile robots currently applied is not too complicated;... Therefore, it can be the more accessible construct of a fuzzy rule base system, and the fuzzy inference mechanism ensures the required control quality.

The performance of a fuzzy controller depends mainly on a fuzzy rule base and fuzzy inference mechanism to handle fuzzy relationships between fuzzy parameters as well as the physical value domain of input-output variables. Therefore, the fuzzy controllers have different capabilities depending on the designed fuzzy rule base. For example, in the calculation of control quantities, the fuzzy controller can calculate and correct some parameters of the system; or additionally calculate the effect of some quantities, such as frictional resistance, external force, and driving force, to compensate for the lack of accuracy of the dynamic model.

Such controllers often integrate crisp control rules with a fuzzy rule base system, for example, the Proportional Derivative-Fuzzy controllers (PDFL), Proportional Integral Derivative-Fuzzy controllers (PIDFL), adaptive fuzzy controllers, and sliding mode fuzzy controllers, ...[28 - 33]. Integrated controllers typically use two or more feedback loops. Fuzzy controllers can also be integrated with other intelligent control rules to form integrated controllers, such as fuzzy neural controllers and hedge algebraic fuzzy controllers [50, 51]. Fuzzy controllers can integrate with intelligent algorithms or use deep learning algorithms and deep reinforcement learning [52 - 56] to determine fuzzy parameters. For example, a genetic algorithm finds the physical value domain of input-output variables. These value domains have a significant influence on the efficiency of the controller.

As shown, the advantage of fuzzy control is that it is simple and easy to implement. Therefore, with knowledge of the control object and natural inference to build a rational rule base system, it is still possible to eliminate the use of dynamic models-fuzzy controllers [27, 57, 62]. Based on the fuzzy rule base system and reasonable inference mechanism have been able to completely replace the dynamic model as well as overcome the uncertainties. However, apart from the above works of the author, there seems to be no other publication on fuzzy control applications for mechanical machining robots that can completely eliminate the use of the dynamic models of the system.

Concerned about the simplicity and ease of implementation of fuzzy control, the article focuses on analyzing the problems that must be solved when designing fuzzy controllers for mechanical machining robots to ensure control accuracy.

The paper also presents the modeling of mechanical machining systems by the robot. From there, it provides insights into the kinematics and dynamics of the robot as the basis for selecting input and output data and partitioning the input and output fuzzy sets of the technological process [12 - 24, 57, 62]. The knowledge about kinematics, especially dynamic characteristics, also plays a vital role in building fuzzy rule base systems, fuzzy relations, and fuzzy inference mechanisms [25 - 27, 57, 59, 62, 67].

After the introduction, the next content of the paper is as follows. Section 2 briefly presents the essential contents of fuzzy sets and fuzzy logic theory, which will be used when designing fuzzy controllers. Section 3 offers the modeling of the technological system, which derives the dynamic model of the robot. The dynamic model, on the one hand, provides a view and understanding of the system's operation, helping to build a fuzzy rule-based system; on the other hand, it is the basis for simulating the operation of the controller. Section 4 presents the robot control system's general structure and analyzes the fuzzy controller's structure and components. Section 5 offers a fuzzy controller applied to mechanical machining robots. As stated, the fuzzy controller designed according to the presented method allows for to completely eliminate of the use of the dynamic model of the system. Section 6 concludes and gives directions for research and development.

2. BRIEF THEORIES

In this section, some content of mathematical modeling of fuzzy set, fuzzy logic related and applied to build a fuzzy controller for the robot is presented briefly [12 - 22].

2.1. Fuzzy Set

Definition. A fuzzy set F on the universe of discourse X is a set whose each element is characterized by a membership function $\mu_F(x)$, which associates each element in X with an actual number whose value is $\mu_F(x)$ in the interval [0, 1] at x, representing the grade of membership of x in F.

Each element of the fuzzy set is an ordered pair (x, $\mu_F(x)$), where $x \in X$, $\mu_F(x)$ is the mapping

$$\mu_F(x): X \to [0,1] \tag{2.1}$$

$$F = \left\{ x, \mu_F(x) \middle| x \in X, \mu_F(x) \in [0,1] \right\} \tag{2.2}$$

Membership function. The membership function of a fuzzy set F determines the membership values of the elements of the universe of discourse (or called the universal set) X in that fuzzy set.

In the form of mathematical models, there have been many proposed membership functions, such as triangular, trapezoidal, and bell-shaped functions, etc. [16-18].

Operations on fuzzy sets.

– Unions (max rule):

$$\mu_{A\cup B}(x) = \max\left\{\mu_A(x), \mu_B(x)\right\} = \mu_A(x) \vee \mu_B(x); \quad \forall x \in X \tag{2.3}$$

– Intersection (min rule):

$$\mu_{A\cap B}(x) = \min\left\{\mu_A(x), \mu_B(x)\right\} = \mu_A(x) \wedge \mu_B(x); \quad \forall x \in X \tag{2.4}$$

Complement:

$$\mu_{\bar{A}}(x) = 1 - \mu_A(x); \quad \forall x \in X \tag{2.5}$$

Fuzzy numbers and intervals. While analyzing real-world systems, the model parameters represent the system, in many cases, uncertain and inaccurate. For example, when calculating $\sqrt{3} \approx 1.732$ or $\sin 45^0 \approx 0.707$. There is often uncertainty and inaccuracy in the measured parameters of machine components such as length, parallelism, perpendicularity, etc. Fuzzy quantities are used to model inaccurate numerical data as linguistically labeled special fuzzy sets whose universal set (defined in the universe of discourse) is the set of real numbers R.

A special fuzzy set, defined on the universal set R of real numbers, is the fuzzy number if it satisfies at least some properties, such as that the fuzzy set is convex, normalized, piecewise continuous, etc.[16 - 18].

2.2. Fuzzy Logic

Fuzzy logic is a generalization of classical logic and deals with problems involving ambiguous and inaccurate information.

Fuzzy logic is logically represented by propositions, which are statements related to some vague concept without crisply defined boundaries; the truth value is any value on the interval [0, 1].

Operators in fuzzy logic

For two simple propositions P_1 and P_2 are defined on fuzzy sets F_1, and F_2, the truth values are p_1 and p_2, respectively, then the fuzzy logic operations are described below.

a) Fuzzy Conjunction

There are four basic t-norms, here two commonly used ones are
Minimum:

$$t_m(P_1 \wedge P_2) = \min(p_1, p_2) = \min(\mu_{F_1}(x), \mu_{F_2}(x)) \tag{2.6}$$

Product:

$$t_p(P_1 \wedge P_2) = p_1 \cdot p_2 = \mu_{F_1}(x) \cdot \mu_{F_2}(x) \tag{2.7}$$

b) Fuzzy Disjunction

There are four basic t-conorms, here two commonly used ones are
Maximum:

$$s_m(P_1 \vee P_2) = \max(p_1, p_2) = \max(\mu_{F_1}(x), \mu_{F_2}(x)) \tag{2.8}$$

Algebraic sum:

$$s_a(P_1 \vee P_2) = p_1 + p_2 - p_1 \cdot p_2 = \mu_{F_1}(x) + \mu_{F_2}(x) - \mu_{F_1}(x) \cdot \mu_{F_2}(x) \tag{2.9}$$

c) Fuzzy Negation

Suppose fuzzy proposition P is assigned to fuzzy set F, the truth value of P denoted p, then the standard negation operation is given by

$$t_{ng}(\overline{P}) = 1 - p \tag{2.10}$$

d) Fuzzy Implication

The truth values of the fuzzy implication of the proposition P implies Q, can be determined as follow

$$t_i(P \to Q) = \neg p \vee q = (1-p) \vee q = \max(1-p, q) \tag{2.11}$$

where p and q are the true values of the propositions P and Q, respectively.

2.3. Fuzzy Relations

When analyzing real-world systems, relations or associations among objects are often studied.

Mathematically, the relation is a general concept and ubiquitous and is formalized from the point of view of set theory. A relation is a mathematical description of an association where certain elements of sets are associated with one another in some way to create the elements of another set.

2.3.1. Definition of Fuzzy Relation

A fuzzy relation R over Cartesian product U1×U2×. . .×Un is any fuzzy subset of U1×U2×. . .×Un, R is defined by a membership function μ_R : U1×U2×. . .×Un → [0, 1].

$$R(x)=\{(x,\mu_R(x))|x=(x_1,\cdots,x_n)\in X_1\times X_2\times\cdots\times X_n,\mu_R(x)\in[0,1]\} \quad (2.12)$$

Suppose F_1, F_2, ..., F_n are fuzzy sets of the universal sets U_1, U_2, ..., U_n, respectively, with $\mu_{Fi}(xi)$, $x_i \in U_i$, i=1,..,n. Then we have the definition:

The fuzzy Cartesian product of the fuzzy sets F_1, F_2,⋯, F_n of U_1, U_2,⋯, U_n, respectively, is the fuzzy relation $F_1 \times F_2 \times \cdots \times F_n$, whose membership function is given by

$$\mu_{F_1\times F_2\times\cdots\times F_n}(x_1,x_2,\cdots,x_n)=\mu_{F_1}(x_1)\wedge\mu_{F_2}(x_2)\wedge\cdots\wedge\mu_{Fn}(x_n), \quad (2.13)$$

where ∧ represents the minimum.

2.3.2. Forms of Representation of the Binary Relations

For A and B are subsets of the universal sets X and Y, respectively, the Cartesian product and fuzzy relation R or R(x,y), on AxB:

$$A\times B=\{(x,y)|x\in A,y\in B\}, \quad (2.14)$$

$$R(x,y)=\{((x,y),\mu_R(x,y))|(x,y)\in A\times B,\mu_R(x,y)\in[0,1]\} \quad (2.15)$$

2.3.3. Operation of Fuzzy Relations

Let R and S be two fuzzy relations on A×B,

$$\begin{aligned}R&=\{((x,y),\mu_R(x,y))|(x,y)\in A\times B,\mu_R(x,y)\in[0,1]\}\\ S&=\{((x,y),\mu_S(x,y))|(x,y)\in A\times B,\mu_S(x,y)\in[0,1]\}\end{aligned} \quad (2.16)$$

Union relation: The union of R and S denoted R∪S is defined as follows

$$\mu_{R\cup S}(x,y)=\max\{\mu_R(x,y),\mu_S(x,y)\}=\mu_R(x,y)\vee\mu_S(x,y), \quad (x,y)\in A\times B \quad (2.17)$$

Intersection relation: The intersection of R and S denoted R∩S is defined as follows

$$\mu_{R\cap S}(x,y)=\min\{\mu_R(x,y),\mu_S(x,y)\}=\mu_R(x,y)\wedge\mu_S(x,y), \quad (x,y)\in A\times B \quad (2.18)$$

Complement relation: The complement of a fuzzy relation R, denoted by $\bar{R}$, is defined as follows

$$\mu_{\bar{R}}(x,y) = 1 - \mu_R(x,y), \quad (x,y) \in A \times B \tag{2.19}$$

2.3.4. Composition of Fuzzy Relations

Let R and S be two binary fuzzy relations in X×Y and Y×Z, respectively. The composition R∘S is a binary fuzzy relation in X×Z whose membership function is defined by

Max-Min composition

$$\mu_{R \circ S}(x,z) = \max_{y \in Y}\{\min(\mu_R(x,y), \mu_S(y,z))\} = \vee_{y \in Y}\{\mu_R(x,y) \wedge \mu_S(y,z)\} \tag{2.20}$$

Max-Product composition

$$\mu_{R \circ S}(x,z) = \max_{y \in Y}\{\mu_R(x,y) \bullet \mu_S(y,z)\} = \vee_{y \in Y}\{\mu_R(x,y) \bullet \mu_S(y,z)\} \tag{2.21}$$

2.4. Linguistic variables

A language variable is a variable whose physical value is represented by words or sentences in a natural or artificial language.

A linguistic value is a natural language term derived using quantitative or qualitative reasoning. A fuzzy linguistic variable x will take values that are linguistic values. Thus, a variable can be represented by physical values and linguistic values. For example, velocity variables can be defined by two value domains. The physical value domain:

$$V = \{x \in R : x \geq 0\} \tag{2.22}$$

The linguistic value domain:

$$^{L}V = \{\text{very slow(VL), slow(L), normal(M), fast(F), very fast(VF)}\} \tag{2.23}$$

here, the universal set is the velocity set V, and the fuzzy set ^{L}V is defined on set V. With a physical value $x \in V$, the membership function can be determined, corresponding to subsets of ^{L}V, and represented as follows:

$$x \to \mu = [\mu_{VL}(x),\ \mu_L(x),\ \mu_M(x),\ \mu_F(x),\ \mu_{VF}(x)]^T \tag{2.24}$$

The above mapping is done during the fuzzification process, in which the physical value of a variable is transformed into linguistic value.

2.5. Fuzzification

Interactive objects of fuzzy operations and fuzzy inference are fuzzy sets. When applying fuzzy theory to analyze real-world systems, it is necessary to represent objects as elements of fuzzy sets.

The process of analyzing a system based on fuzzy theory can be encapsulated in three main steps:

1) Representing objects on the fuzzy set form defined on the universe of discourse (universal set);
2) Effects on objects (fuzzy sets) by fuzzy operations and fuzzy inference results in newly created fuzzy sets;
3) Transform new fuzzy sets (results of fuzzy operations and fuzzy inference) of a universal set into variables with physical values.

These steps are called Fuzzification, the Fuzzy rule base and Inference Mechanism (Fuzzy Composition Device), and Defuzzification, respectively.

Fuzzification is the step where the system's considered objects (inputs) are modeled by fuzzy sets with their respective domains. In the fuzzification process, there are central issues to be considered:

1) The identification of the input and output signals and the corresponding physical value domains.
2) Language variables and scale mapping of input and output data.
3) Strategy for filtering and selecting input data.
4) Membership values Assignments.

These issues are discussed further in section 4.2.

2.6. Fuzzy Rule Base and Inference Mechanism

The fuzzy rule base contains the fuzzy controller rules generated based on the knowledge of the controlled object.

The fuzzy rule (Fuzzy controller rule, also known as fuzzy implication) is a fuzzy relation between propositions according to the IF-THEN rule.

Fuzzy Rules Single-Input/Single-Output

$$R_i : \; If \;\; q \; is \; A_i \;\; then \; u \; is \; C_i \tag{2.25}$$

A_i and C_i are linguistic values defined by fuzzy sets on the universe of discourse Q and U, respectively. The proposition “q is Ai” is called the “antecedent” or “premise,” while “u is Bi” is called the “consequence” or “conclusion.” The rule is also called a “fuzzy implication” or “fuzzy conditional statement.”

The expression (2.25) describes a relation between two variables, q and u. This suggests that a fuzzy rule can be defined as a binary relation R on the product space $Q \times U$.

Fuzzy Rules Two-Input/Single-Output

$$\begin{cases} \quad\;\; R_1 : \; If \;\; q \; is \; A_1 \;\; and \;\; v \; is \; B_1 \;\; then \; u \; is \; C_1 \\ else \; R_2 : \; If \;\; q \; is \; A_2 \;\; and \;\; v \; is \; B_2 \;\; then \; u \; is \; C_2 \\ \dots \\ else \; R_n : \; If \;\; q \; is \; A_n \;\; and \;\; v \; is \; B_n \;\; then \; u \; is \; C_n \end{cases} \tag{2.26}$$

where q, v and u are linguistic variables representing the state variables and the control variables, respectively. A_i, B_i, and C_i are linguistic values of the linguistic variables q, v, and u in the universe of discourse Q, V, and U, respectively, for i=1, 2, …, n.

Each fuzzy control rule R_i is implemented as a fuzzy implication relation and is defined as

$$R_i : \; (A_i \; and \; B_i) \; \to \; C_i \;\; \text{or} \tag{2.27}$$

$$\begin{aligned} \mu_{R_i} &= \mu_{(A_i \; and \; B_i \to C_i)}(q,v,u) \\ &= \left[\mu_{A_i}(q) \; and \; \mu_{B_i}(v) \right] \to \mu_{C_i}(u) \end{aligned} \tag{2.28}$$

where “A_i and B_i” is a fuzzy set $A_i \times B_i$ in $Q \times V$; R_i: (A_i and B_i) $\to$ C_i is a fuzzy implication relation in $Q \times V \times U$, and $\to$ denotes a fuzzy implication function.

The number of premise propositions, result propositions, and fuzzy rules (2.25) or (2.26) depend on the control object and the designer's expert knowledge.

The basic fuzzy rules are simple and naturally compatible with the real system. The problem is how to use these rule base systems (2.25) or (2.26) to determine the compositional rules for the controller.

These issues and the construction of the fuzzy rule system for the controller will be further presented in Section 4.3.

2.7. Defuzzification

Defuzzification is the mapping process that converts fuzzy results into physical (crisp) values. In practical applications, the control commands operate according to the physical value. Therefore, it is necessary to defuzzify the results of fuzzy inference. There are many methods for defuzzification techniques. This issue is further presented in section 4.4.

3. MODELLING OF MACHINING ROBOT IN MECHANICAL PROCESSING

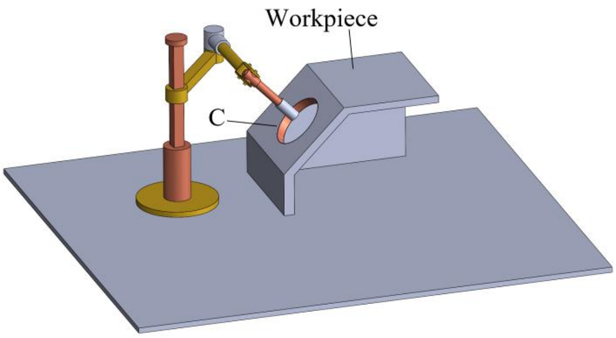


Figure 3.1. Open-chain serial structure robot that performs machining by grinding.

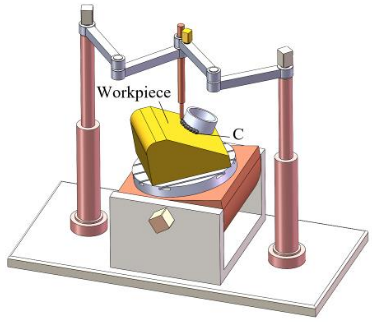


Figure 3.2. Closed-chain serial structure robot, performing arc welding.

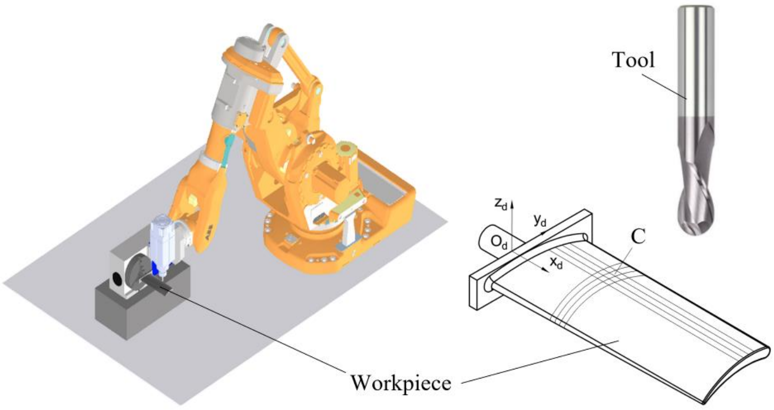


Figure 3.3. Robot for finishing milling or surface grinding.

Technology-manipulating robots, in general, and mechanical processing robots, in particular, have a common structure consisting of many links connected by joints to form the chains, called the kinematic chains of the robot. In classification, this type of robot is called a manipulator [93 - 99]. The kinematic sequences of the robot can be open-chain; or closed-chain [62, 93 - 96]. The robot's technological operations in general and mechanical machining operations, collectively referred to as machining operations, are achieved by the movement of the manipulator. The processing tool is mounted on the end-effector link, moving with the end-effector. According to the technical requirements of the machining operation, the manipulation movement of the robot must follow the law so that the tool moves along the trajectory and interacts with the workpiece. The motion of the manipulator is synthesized from the movement of the links driven by motors located at the active joints. Some example models of mechanical machining robots are shown in Figures 3.1, 3.2, and 3.3. Figure 3.1 shows an open-chain serial structure robot that performs machining by grinding. Figure 3.2 shows a closed-chain serial structure robot performing arc welding, grinding, or deburring, etc. Figure 3.3 shows a robot for finishing milling or surface grinding. The tool on the end-effector executes the operation motion so that the end impact point moves along the C-curve, as shown in the figures while ensuring the direction of the tool when acting on the workpiece.

Controlling the mechanical processing robot is to control the motor to drive the links so that it can be synthesized into the end-effector's movement according to the machining technique's requirements.

The motion and motion relationship between the end-effector and middle links are determined by law, through kinematic modeling, as input to the controller.

3.1. Kinematic Modelling

In mechanical engineering, a tool path is constructed to determine the law of motion of the tool and is defined in a coordinate system, usually a base coordinate system. A coordinate system attached to the tool, which characterizes the tool's motion and the end-effector, is referred to as the tool coordinate system. Thus, the position of the tool coordinate system in the base coordinate system is determined by the position and orientation coordinates, called the operation coordinates (3.1) [50, 51, 57, 58, 62].

$$p = {}^0p_E = \left[{}^0p_1, {}^0p_2, ..., {}^0p_6\right]^T = \left[{}^0x_E, {}^0y_E, {}^0z_E, {}^0\alpha_E, {}^0\beta_E, {}^0\eta_E\right]^T \quad (3.1)$$

In (3.1), the first three elements are the position coordinates of the coordinate system origin, and the remaining elements are the direction coordinates. The symbol ($_E$) in the lower right corner is for the instrument symbol. The index (^{0}) in the upper left corner indicates that the parameters are represented in the base coordinate system. In calculations, coordinate transformations and coordinate transformation matrices are often used. Accordingly, the position and orientation of the tool are also represented by a homogeneous coordinate transformation matrix, symbol ${}^0A_E(p)$. The matrix ${}^0A_E(p)$ can be determined by the operation coordinates p, or vice versa from the matrix ${}^0A_E(p)$ can be determined p (3.2) [97-99].

$${}^0A_E(p) = \begin{bmatrix} {}^0C_E\left({}^0\alpha_E, {}^0\beta_E, {}^0\eta_E\right) & {}^0r_E\left({}^0x_E, {}^0y_E, {}^0z_E\right) \\ 0^T & 1 \end{bmatrix} \quad (3.2)$$

Based on the robot's kinematics structure, coordinate systems are attached to the robot's links, and homogeneous coordinate transformation matrices are used to represent the transformation between coordinate systems [57, 62, 95 - 99].

Matrix symbol ${}^{i-1}A_i(q_i)$ represents the position of the "i" coordinate system of link "i" relative to the "i-1" coordinate system of link "i-1", q_i is the joint coordinate representing the motion of link "i" with respect to link "i-1". The motion of the robot's links is represented by the vector of joint coordinates (3.3).

$$q = \left[q_1, .., q_n\right]^T \quad (3.3)$$

where n is the number of joint coordinates of the robot.

Many methods are applied to transform coordinates, such as the Denavit-Hartenberg method, John Craig method, generalized coordinate method, Quaternion method, and Dual quaternion, etc.

From the coordinate transformations, the matrix determining the position of the tool concerning the base coordinate system can be calculated according to (3.4).

$$^{0}A_{E}(q) = {}^{0}A_{1}(q_{1})\,{}^{1}A_{2}(q_{2})\ldots{}^{n-1}A_{n}(q_{n})\,{}^{n}A_{E} = \begin{bmatrix} {}^{0}C_{E}(q) & {}^{0}r_{E}(q) \\ 0^{T} & 1 \end{bmatrix} \quad (3.4)$$

Given the required machining motion of the tool and the end-effector, the operation coordinates and their derivatives concerning time (determined in terms of velocity and acceleration) are determined by (3.1) or (3.2). From (3.2) and (3.4), the laws of motion of the links are also determined. For example, the motion of the robot's links is represented by the joint coordinates and the time derivative of the joint coordinates, as shown in (3.5).

$$\begin{cases} q(t) = \left[q_{1}(t),..,q_{n}(t)\right]^{T} \\ \dot{q}(t) = \left[\dot{q}_{1}(t),..,\dot{q}_{n}(t)\right]^{T} \\ \ddot{q}(t) = \left[\ddot{q}_{1}(t),..,\ddot{q}_{n}(t)\right]^{T} \end{cases} \quad (3.5)$$

The motion law defined by (3.5) is the database for the input of the robot control system. The controller calculates the force/torque driving the motors to ensure the desired motion law.

3.2. Dynamic Modelling

The forces/torque driving the motors, referred to as the control force for short, is determined based on the laws of motion and the dynamic characteristics of the robot.

Crisp controllers for industrial robots often use robot dynamics [50, 51, 57, 62, –97-99]. Intelligent controllers such as fuzzy controllers, hedge algebraic controls, neural network controls, etc., can use approximate dynamic models or do not need dynamic models depending on the capabilities of the designed controller.

As mentioned, mechanical machining robots usually have open-chain or closed-chain serial structures. For an open-chain structure robot, the differential equation of motion of the robot is the Lagrange equation in the matrix form below (3.6) [50, 51, 57, 97 - 99].

$$M(q)\ddot{q} + \psi(q,\dot{q}) + G(q) + Q = U \quad (3.6)$$

where, $M(q) \in R^{n\times n}$ is the mass matrix of the robot system, calculated through mass m_i, translation Jacobian matrix $J_{Ti} \in R^{3\times n}$, rotation Jacobian matrix $^{ci}J_{Ri} \in R^{3\times n}$ and inertia tensor $^{Ci}\Theta_{Ci}$ $\in R^{3\times 3}$ of moving links “I” (i = 1,..,n) (3.7), (3.8).

$$M(q) = \sum_{i=1}^{n}\left(J_{Ti}^{T} m_{i} J_{Ti} + {}^{ci}J_{Ri}^{T}\,{}^{ci}\Theta_{ci}\,{}^{ci}J_{Ri}\right) \quad (3.7)$$

$$J_{Ti} = \frac{\partial r_{ci}}{\partial q}; \qquad {}^{ci}J_{Ri} = \frac{\partial\,{}^{ci}\omega_{i}}{\partial \dot{q}} \quad (3.8)$$

$\psi(q,\dot{q}) \in R^n$ is the vector of the generalized forces of Coriolis and centrifugal inertial forces, calculated through the partial derivative of the elements of matrix M(q) with respect to the generalized coordinates q and the generalized velocity $\dot{q}$ (3.9).

$$\psi(q,\dot{q}) = \left[\psi_1, \psi_2, .., \psi_n\right]^T; \ \psi_j = \sum_{k,l=1}^{n} (k,l;j)\dot{q}_k \dot{q}_l; \ (k,l;j) = \frac{1}{2}\left(\frac{\partial m_{kj}}{\partial q_l} + \frac{\partial m_{lj}}{\partial q_k} - \frac{\partial m_{kl}}{\partial q_j}\right) \quad (3.9)$$

(k,l;j) is Christoffel notation with 3 indexes of the first kind; m_{kl} (k,l = 1,…,n) are the elements of mass matrix M(q).

G(q) $\in R^n$ is the vector of the generalized forces of the conservative forces (3.10).

$$G(q) = \left[G_1, G_2, .., G_n\right]^T; \quad G_j = \frac{\partial \Pi}{\partial q_j} \quad (3.10)$$

where Π is the total potential energy of the system.

Q(q) $\in R^n$ is the vector of the generalized forces of the non-conservative forces, including the friction forces, cutting forces, etc. (3.11), (3.12) [57].

$$Q(q) = J_{TE}^T F_E + J_{RE}^T M_E = \left[J_{TE}^T, J_{RE}^T\right]\left[F_E, M_E\right]^T = J_E^T R_E \quad (3.11)$$

$$J_{TE} = \frac{\partial r_E}{\partial q}; \quad J_{RE} = \frac{\partial \omega_E}{\partial \dot{q}} \quad (3.12)$$

where, $F_E \in R^3$, $M_E \in R^3$ – are the sum of non-conservative forces and moments acting on the tool of the end-effector, respectively; $r_E \in R^3$, $\omega_E \in R^3$ – respectively is the position of the point of application of force F_E, and the angular velocity of the end-effector E under the effect of the total moments M_E; U $\in R^n$ is the vector of the generalized forces of the driving forces/torques for the motors (3.13).

$$U = \left[U_1, U_2, .., U_n\right]^T; \quad U_i = \tau_i \quad (3.13)$$

τ_i – is the driving force/torque of the "I" motor.

With the open-chain serial structure robot, when applying the dynamic model for control, the equations from (3.6),..., and (3.13) allow the convenient calculation to determine the control force. The equation in matrix form (3.6) includes n independent motion differential equations corresponding to n degrees of freedom of the robot.

With closed-chain kinematic structure robots, the calculation is more complicated. The system of differential equations of motion of the robot is the Lagrange equation with the multipliers of the form (3.14) [62,63,94,96].

$$\begin{cases} M(q)\ddot{q} + \psi(q,\dot{q}) + G(q) + Q + Q^c = U & (3.14.a) \\ f^c(q) = 0 & (3.14.b) \end{cases} \quad (3.14)$$

The equation (3.14.b) is composed of s kinematic constraint equations established from the conditions of the closed kinematic loops. With the robot structure under consideration, the constraint equations (3.14.b) are holonomic, which can be represented in the form of algebraic equations. With a suitable initial condition, (14. b) can be reduced to a differential form for a convenient investigation of the overall system of equations (3.14).

$Q^C \in R^n$ is the vector of the generalized forces of the constraint reactions caused by constraints (3.14.b). By holonomic constraints, Q^C can be calculated as follows (3.15) [62, 94, 96].

$$Q^c = J(q)^T \lambda; \quad J(q) = \frac{\partial f^c}{\partial q} \tag{3.15}$$

In which the components of the vector λ are the Lagrange multipliers.

$$\lambda = [\lambda_1, ..., \lambda_s]^T \tag{3.16}$$

The other components in (3.14) are calculated and expressed according to the formulas in form, similar to (3.7) ÷(3.13).

The system of equations (3.14) is reduced to the form (3.17) consisting of n+s differential equations with n generalized coordinates of q and s Lagrange multipliers of λ, that is, there are n+s unknowns. Thus, the system of equations (3.17) is solvable and can be applied to the robot controller [62, 94, 96].

$$\begin{cases} M(q)\ddot{q} + \psi(q,\dot{q}) + G(q) + Q + J(q)^T \lambda = U & (3.17.a) \\ J(q)\ddot{q} = \dot{J}(q,\dot{q})\dot{q} & (3.17.b) \end{cases} \tag{3.17}$$

Robot dynamics models from (3.6) to (3.13) or from (3.7) to (3.17) are applied when implementing crisp controllers. On the other hand, understanding the dynamics of the robot based on the given dynamic models will support the design of the fuzzy controller, building the fuzzy rule base system.

4. ROBOT FUZZY CONTROL

In this section, the general structure of the robot control system is presented, and the components of the fuzzy controller are analyzed. The first elements of fuzzy controller design are determining input and output variables, scale mapping the physical value of input and output data to the partitioned universe of discourse and converting them to corresponding language values defined and represented by fuzzy sets. The following important task is to build a fuzzy rule base system, fuzzy inference mechanism, and analysis and processing algorithms; these are discussed. Finally, the defuzzification method commonly applied in published studies and suggested in this study is also introduced.

4.1. The General Structure of the Robot Control System

Figure 4.1 shows the structure of a robot control system, which is a standard structure for both crisp controllers as well as intelligent controllers such as Fuzzy Logic Control (FLC); Hedge Algebrass Control (HAC); Neural Network Control (NNC),... The main difference between control systems is the "Control" block. The building blocks of the control system are [16 - 22, 25 - 27]:

Input block – is the input signal block (input data-INT DATA). With a crisp controller, this block has data about the robot's motion trajectory, including position, velocity, acceleration, external forces interacting with the robot, and possibly other influencing factors [123 - 125].

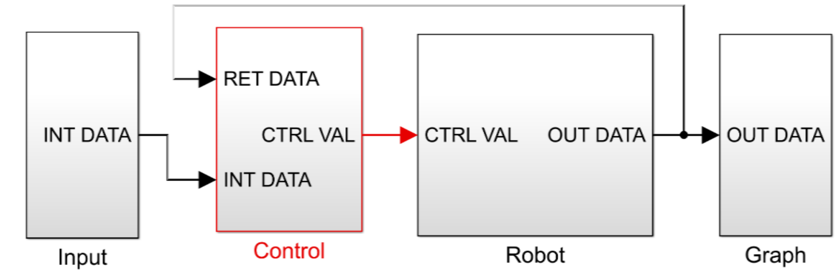


Figure 4.1. Structure of the robot control system.

With FLC, the INT DATA block consists only of files containing data about the position and velocity of the motion, without the need for data on acceleration. There is also a data block in the form of a matrix containing the physical value domain of language variables. Depending on the capabilities of the designed fuzzy controller, external forces, noise, and other influencing factors may appear to be approximate or unnecessary [27, 57, 62].

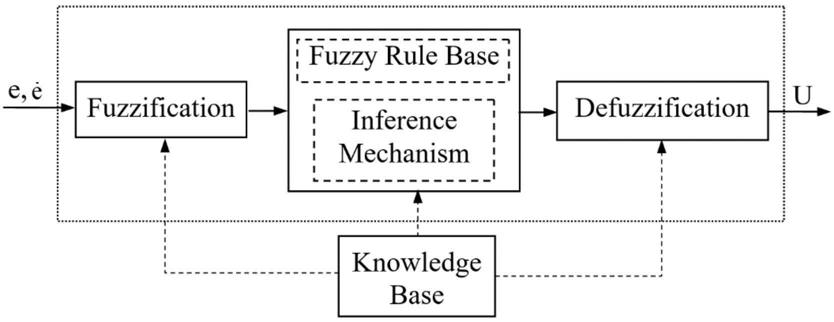


Figure 4.2. Structure of robot controller based on fuzzy logic.

Control block - In general, a control block is a block that uses an algorithm based on control laws to calculate control quantities. In robot control, the control quantities are force/torque (control force). With a crisp controller, the control block calculates the driving force based on the dynamic model of the robot and the crisp control law [97 - 99].

In the FLC, the Control block has the structure shown in Figure 4.2. There are components including Fuzzification, Fuzzy Composition Device (Fuzzy Rule Base and Fuzzy Inference Mechanism), Defuzzification, and the Expert block (Knowledge Base) manages the entire control block [25, 69, 97 - 99].

Robot block – a controlled object receives control signals from the Control block and acts to control the robot to perform manipulation movements according to the predetermined trajectory. In the control system simulation program, this block is the forward dynamics algorithm to calculate the motion to determine the robot's operating state under the action of the driving force.

Graph block – displays system performance results.

4.2. Fuzzification

As pointed out, the fuzzy controller works by natural inference based on linguistic variables and linguistic semantic values. Therefore, the system's investigated objects (inputs, outputs) must be modeled as fuzzy sets of linguistic variables with their respective domains. That's called fuzzification. Fuzzification is unnecessary if fuzzy sets or relations represent the objects (input, output).

The Fuzzification process is carried out according to the following principal contents.

4.2.1. The Identification of the Input and Output Signals and the Corresponding Physical Value Domains

Like the crisp controller, the identification of the input and output signals of the fuzzy controller depends on the object to be controlled, first of all, the input signals that reflect the state of the system, such as signal to determine position, velocity, ...Next are the output signals, which are the result of decision making by the controller based on the input signals, for example, the driving force that ensures the desired operating state (position, velocity,…) of the controlled system. The problem with fuzzy controllers is that the input and output signals can be imprecise but must be defined by a physical value domain. The reasonably determined physical value domain plays a vital role in ensuring the accuracy of the fuzzy controller, but this issue is rarely considered, presented, or discussed in the studies.

Usually, the physical value domain of the linguistic variables representing the input and output signals is determined within the range bounded by two minimum and maximum values (4.1).

$$v \in \left[v_{\min}, v_{max} \right] \tag{4.1}$$

The determination of the value domain (4.1) is based on the knowledge of the controlled object and the knowledge and experience of the expert. This is often done randomly in published studies, with little analysis or correction [16 - 18, 57 - 81].

In this work, the selection and correction of the domain of reasonable physical values are of particular interest, and thus the fuzzy controller gives good results.

Thus, to determine (4.1) is reasonable, it can be tested and corrected by simulation and experimental methods. In addition, it is possible to apply intelligent algorithms such as deep learning, reinforcement learning, deep reinforcement learning, genetic algorithms, modern evolutionary algorithms, etc.

The physical value domain and the fuzzy rule base and inference are the main factors determining the performance and quality of the fuzzy controller. This issue will be discussed further in section 4.3 when presenting the construction of a fuzzy rule base system and fuzzy inference mechanism.

4.2.2. Language Variables and Scale Mapping of Input and Output Data

The algorithm of the crisp controller works with the numerical values of the controller's parameters. In the same way, the fuzzy controller works based on fuzzy operations with the linguistic semantic values of the fuzzy controller variables. That raises the problem of quantifying such quantities for digital computers.

Therefore, the next step of fuzzification is naming the linguistic variables of the input and output signals and fuzzy partitioning. Fuzzy partitioning uses scale mapping to partition the universe of discourse of linguistic variables to establish fuzzy sets and to associate linguistic variables with term sets of corresponding fuzzy sets [17].

Partition of the universe of discourse, known as quantization, divides the universe into a certain number of segments. Each segment is labeled and forms a discrete universe. Then the corresponding fuzzy sets are determined on the discrete universe of discourse. Next, scale mapping converts the crisp values and the physical domain of the input and output signals into the fuzzy set with the corresponding linguistic semantic value. Depending on the control object, the scale mapping can be uniform (linear), heterogeneous (nonlinear), or both.

In a partitioned fuzzy set, the fuzzy set label or corresponding linguistic term determines the maximum value of the membership function. For example, the universe of discourse for the velocity variable is divided into three partitions and defines three fuzzy sets Small, Medium, and Big. At the same time, the names of those established fuzzy sets are used as the set of linguistic terms associated with the linguistic variable. Thus, the membership function value at Small, Medium, and Big are μ_F(Small) = 1, μ_F(Medium) = 1, μ_F(Big) = 1.

Thus, the number of fuzzy sets formed from the partitioned universe of discourse is also the number of linguistic terms of the linguistic variable. Therefore, when calling the number of linguistic terms assigned the semantic value of a linguistic variable, it is also the number of partitions or quantization levels of the discourse universe.

Usually, the number of quantization levels of the discourse universe (the number of linguistic terms assigned linguistic values) should be large enough to obtain a suitable approximation. However, the accuracy is not proportional to the number of quantization levels. The linguistic terms used in published studies can be from 3, 5, 7, and 9. The most common are 5 and 7 terms [17, 18, 26, 27, 31, 43, 51, 58, 69, 79].

Table 4.1 shows the partitions of the partitioned universe of discourse X, fuzzy sets, linguistic value terms, and their corresponding symbols, with a quantization level of 5. Similarly, Table 4.2 represents a quantization level of 7.

Table 4.1. The partitions, fuzzy sets, linguistic terms, and symbols correspond to the universe set X quantization level of 5.

No.	*Levels X_i*	*Fuzzy Set Fi*	*Name of Linguistic Value*	*Notation of Linguistic Value*
1	X_1	F_1	Negative Big	NB
2	X_2	F_2	Negative Small	NS
3	X_3	F_3	Zero	Z
4	X_4	F_4	Positive Small	PS
5	X_5	F_5	Positive Big	PB

Table 4.2. The partitions, fuzzy sets, linguistic terms, and symbols correspond to the universe set X quantization level of 7.

No.	*Levels X_i*	*Fuzzy Set Fi*	*Name of Linguistic Value*	*Notation of Linguistic Value*
1	X_1	F_1	Negative Big	NB
2	X_2	F_2	Negative Small	NS
3	X_3	F_3	Negative	N
4	X_4	F_4	Zero	Z
5	X_5	F_5	Positive	P
6	X_6	F_6	Positive Small	PS
7	X_5	F_7	Positive Big	PB

4.2.3. Strategy for Filtering and Selection of Input Data

Input variables can be obtained from observations or measurements as crisp values. These values are probabilistic because they can be biased by random noise. As pointed out in [17], for the convenience of computation by fuzzy operations, the input data should be converted to fuzzy numbers since fuzzy numbers are much easier to manipulate than random variables. The input signal can be considered a singleton if the input data does not contain many random values but is determined by the exact crisp value. Therefore, the input data can be in the form of a fuzzy set or a singleton.

4.2.4. Membership Values Assignments

As is known, to perform fuzzy operations, the input data must be represented in the form of membership function values belonging to one or more fuzzy sets partitioned on the corresponding universe of discourse.

Different methods exist for assigning membership values or functions to language variables. Many different types of membership functions have been applied in the control engineering of engineering systems. The most commonly used membership functions in robot controllers are triangular and trapezoidal [17, 18, 26, 27, 51, 57, 58]. These forms are simple and easy to calculate, ensuring the controller's performance and the required accuracy.

4.3. Fuzzy Composition Device

The Fuzzy Composition Device is the heart of a fuzzy controller; it includes a fuzzy rule base system, a fuzzy inference mechanism, and integrated, processed for making decisions by algorithms. Thus, two main problems determine the performance of this device's performance: the method of building a fuzzy rule base (fuzzy implication method) and the method of composition of fuzzy compositional rules.

4.3.1. Fuzzy Rule Base System

As is known, the fuzzy rule, also called fuzzy implication, is a fuzzy relation between one proposition to another proposition, according to the IF-THEN rule, as pointed out in Section 2 (2.26).

In (2.26), the consequential propositions (after "then") are the results implied by the fuzzy relations between the antecedent propositions (before the "then") and the fuzzy relations between these propositions with consequential propositions. The propositions before "then" in (2.26) (also known as Conditions or Antecedents) represent the input variables, also called state variables, of the system and depend on the properties of the system. The proposition after "then" (conclusions or consequences) is decided based on the expert knowledge and experience of the designer about the controlled object [17, 18, 26, 27, 51, 57, 58, 62,...].

The number of the antecedent and consequential propositions in the IF-THEN rule (2.26) depends on the number of input and output signals of the language variables, therefore, on the control object. Thus, the number of IF-THEN rules (2.26) of the rule base system depends on the number of input signals of the linguistic variables and the number of fuzzy sets of those linguistic variables.

The robot controller is often required to ensure the robot follows the trajectory. Therefore, the robot controller usually has two input signals of state variables: position and velocity, and one output signal of the control variable, the control force.

Let the input and output signals of the linguistic variables be position error e(t), velocity error $\dot{e}(t)$ (input signal), control force u(t) (output signal), and denote the corresponding universes of discourse for these variables as Q, V, and U.

Assuming that the universe of discourse Q, V is partitioned as shown in Table 4.1 or Table 4.2, the rule base system (2.26) for the robot controller can be created as below (4.2):

$$\begin{cases} \text{If } e(t) \text{ is NB and } \dot{e}(t) \text{ is NB then } u(t) \text{ is PB or} \\ \text{If } e(t) \text{ is NB and } \dot{e}(t) \text{ is NS then } u(t) \text{ is PB or} \\ \vdots \\ \text{If } e(t) \text{ is PB and } \dot{e}(t) \text{ is PB then } u(t) \text{ is NB} \end{cases} \tag{4.2}$$

In general, (4.2) can be written in the form (4.3)

$$\begin{aligned} &R_i: \textit{If } e(t) \textit{ is } F_{e,k} \textit{ and } \dot{e}(t) \textit{ is } F_{de,l} \textit{ then } u(t) \textit{ is } F_{u,j} \\ &k = 1, 2, \ldots, m;\ l = 1, 2, \ldots, n;\ j = 1, 2, \ldots, p;\ i = 1, \ldots, N = m \times n. \end{aligned} \tag{4.3}$$

in which, $F_{e,k}$, $F_{de,l}$, $F_{u,j}$ are fuzzy sets defined on the universes of discourse Q, V, U respectively; Indices k, l, j represent the fuzzy component sets; m, n, p - the number of fuzzy sets; N is the number of base rules (4.3). Sometimes the representation of "$\dot{e}$" is not convenient because the symbol in the formula is too small, so "de" represents "$\dot{e}$".

$$F_{e,k} \subset Q,\ F_{de,l} \subset V,\ F_{u,j} \subset U$$

Based on Tables 4.1 and 4.2, with the cases of choosing different quantization levels for universes of discourse, the rule base system (4.2) and (4.3) can have a rule number of 25, 35, 35, and 49.

Thus, the consequence proposition in (4.2) and (4.3) is decided by the designer and depends on the number of fuzzy sets partitioned for the output variable, i.e., the control variable. The designer selects the consequence proposition of (4.2), essentially selecting the partitioned fuzzy set of the corresponding control variable for each fuzzy rule (4.2).

Assuming the quantization level is N=mxn=5x5=25, the rules are defined and summarized in a fuzzy associative matrix (FAM) table, Table 4.3 [17, 18, 26, 27, 51, 57, 58]. Thus, the values in the FAM table represent the control outputs.

Table 4.3. Fuzzy rule base system.

U(*t*)	Position error *e*(*t*)				
Velocity Error $\dot{e}$ (*t*)	**NB**	**NS**	**Z**	**PS**	**PB**
NB	PB	PB	PB	PS	Z
NS	PB	PS	PS	Z	NS
Z	PB	PS	Z	NS	NB
PS	PS	Z	NS	NS	NB
PB	Z	NS	NB	NB	NB

The aggregation rule of the fuzzy rule base system for a fuzzy controller is as follows.

$$\begin{aligned}
&R:\ \bigcup_{s=1}^{N} \mathrm{R}_i \\
&R_i:\ \textit{If}\ \ e(t)\ \textit{is}\ F_{e,k}\ \textit{and}\ \dot{e}(t)\ \textit{is}\ F_{de,l}\ \textit{then}\ u(t)\ \textit{is}\ F_{u,j},\ \ i=1,...,N \\
&\mu_{R_i}=\mu_{\left(F_{e,k}\ \textit{and}\ F_{de,l}\rightarrow F_{u,j}\right)}\left(e,\dot{e},u\right) \\
&\quad =\left[\mu_{F_{e,k}}(e)\ \textit{and}\ \mu_{F_{de,l}}(\dot{e})\right]\rightarrow\mu_{F_{u,i}}(u)
\end{aligned} \tag{4.4}$$

The problem is how to use the rule base system (4.2) to determine the control variable (output signal) based on the state variable (input signal). To do that, an inference operator is built to generate an aggregate rule called a fuzzy compositional rule of inference.

4.3.2. Fuzzy Compositional Rule of Inference

Although there are various methods for aggregating fuzzy compositional rules, as mentioned in section 2, Mamdani's approach demonstrates simplicity and ease of implementation, which guarantees the performance and accuracy of the controller. Therefore, Mamdani's method is widely applied in control engineering in general and robot control [17-22]. In particular, the works [27, 57, 62] show that in the case of mechanical processing robots, high requirements for control accuracy are still ensured when applying the Mamdani method.

Mamdani's method is applied to both the Min operator (MIN) for fuzzy implication and the MAX-MIN operator for aggregation.

The time variation of the input signals of the position error e(t) and the velocity error $\dot{\mathrm{e}}(\mathrm{t})$ are represented by F'_{e} and $\mathrm{F}'_{\mathrm{de}}(\mathrm{de}=\dot{\mathrm{e}})$, respectively. Here F'_{e}, $\mathrm{F}'_{\mathrm{de}}$ can be crisp values or fuzzy sets. With the robot controller, the input signals are the feedback signals determined by the sensors, and so F'_{e}, $\mathrm{F}'_{\mathrm{de}}$ are physical values. Thus the input signal will be singleton [17, 27, 57, 62].

Assuming the fuzzy rule base system (4.2) has N rules R_i (i=1,.., N), and for input signals $\mathrm{F}'_{\mathrm{e}}, \mathrm{F}'_{\mathrm{de}}$, then the fuzzy set F'_{u} of the output linguistic variable and its membership function $\mu_{\mathrm{F}'_{\mathrm{u}}}$ is determined by the following fuzzy relation:

$$\begin{cases} F'_{u,i} \ = \ \left(F'_e,\ F'_{de}\right)\circ R_i\ ; \quad (de=\dot{e}) \\ F'_u=\bigcup_{i=1}^{N} F'_{u,i} \end{cases} \tag{4.5}$$

Mamdani's MIN operator is used for implication:

$$\mu_{F'_{u,i}}(u)=\alpha_i\wedge\mu_{F_{u,i}}(u);\quad \alpha_i=\mu_{F_{e,k}}\left(F'_e\right)\wedge\mu_{F_{de,l}}\left(F'_{de}\right) \tag{4.6}$$

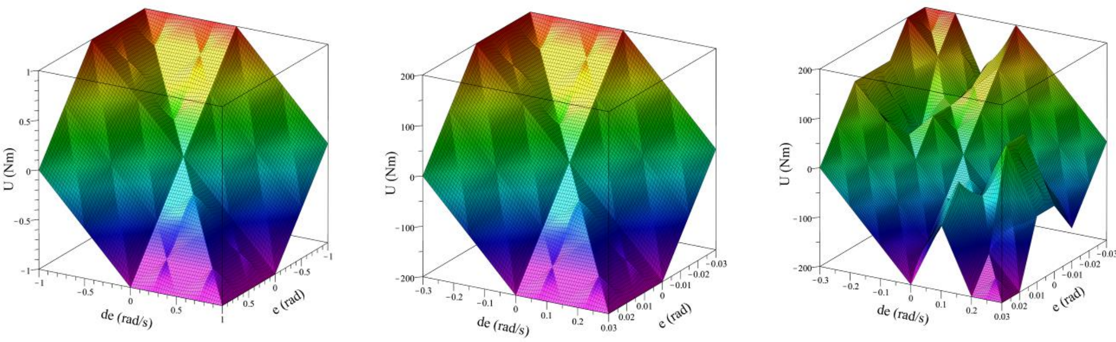
where, i - index of fuzzy rule i; k, l - is the index of the corresponding fuzzy sets $F_{e,k}$, $F_{de,l}$ to which the input signals $\mathrm{F}'_{\mathrm{e}}, \mathrm{F}'_{\mathrm{de}}$ belong; $\mu_{F_{u,i}}(u)$ is the membership function defining the fuzzy implication relation of the fuzzy rule R_i, as shown in (4.4).

The compositional rule of the entire fuzzy rule base system is determined by the membership value based on Mamdani's MAX-MIN operator:

$$\mu_{F'_u}(u)=\bigvee_{i=1}^{N}\left[\alpha_i\wedge\mu_{F_{u,i}}(u)\right]=\bigvee_{i=1}^{N}\mu_{F'_{u,i}}(u) \tag{4.7}$$

4.3.3. Fuzzy Control Hypersurfaces

This part simulates the controller's compositional fuzzy rule of inference based on the physical value domain, the fuzzy rule base system, and the fuzzy inference mechanism. This helps to visualize the influence of the physical value domain and the rule base system and helps to make corrections.



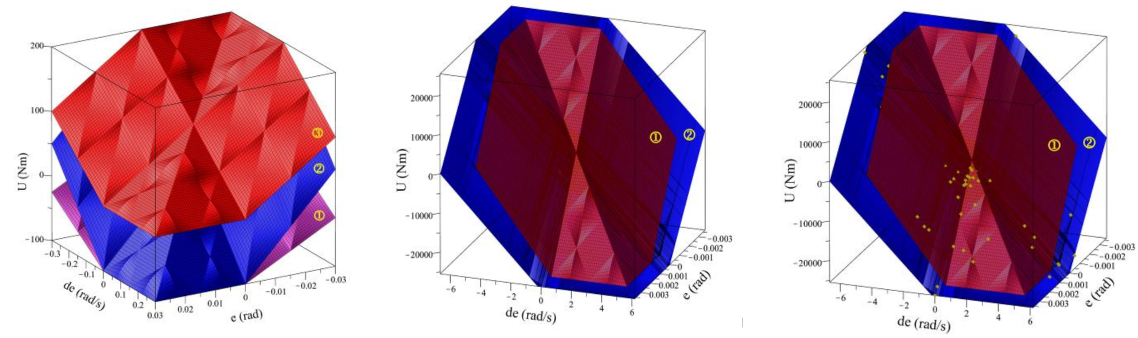
4.3.a 4.3.b 4.3.c

Figure 4.3. The grid hypersurfaces represent the relationship between input state variables and output control variables (Control Surfaces).

In general, the relationship between input state variables and output control variables can be represented by grid hypersurfaces, referred to as control surfaces. The robot controller has two input state variables and one output control variable, so the mesh hypersurface is represented in Euclidean space, as shown in Figure 4.3. Figure 4.3.a shows the control surface in which the universes of discourse are transformed into the normalized closed interval [-1, 1]. Figures 4.3.b and 4.3.c show control surfaces in which the universes of discourse are partitioned on the physical value domain, the same physical value domain but with different fuzzy rule base systems. Control surfaces, as Fig. 4.3.a, are often used to observe and study when building a fuzzy rule base system and determining the physical value domain of input and output variables. It is observed that the geometrical shapes of the control surfaces corresponding to the universes of discourse are normalized or defined over the physical value domain are similar. Control surfaces 4.3.b and 4.3.c are often used for numerical simulation and numerical experimentation with fuzzy rule base systems and defined physical value domains of input and output variables to perform corrections.

While there are different types of membership functions, the most commonly used are the triangle and trapezoid functions [17, 18, 26, 27, 51, 57, 58, 62,..]. The reason for this may be the simplicity of the calculation. On the other hand, for technical systems in general and robots, the triangular and trapezoidal functions are suitable for the system's dynamic characteristics. Therefore, only the above two membership functions are applied here.

Figure 4.4 shows control surfaces with different physical value domains of input and output variables. Figure 4.4.a shows the control surfaces with the same physical value domains of the input variables but different outputs. Figure 4.4.b shows control surfaces with different physical value domains of the input variables, in which the control surface ① corresponds to the estimated physical value domain when designing, and the control surface ② corresponds to the actual physical value domain when simulating the operation of the controller. Figure 4.4.c differs from Figure 4.4.b by adding a dotted surface, where each dot represents a relationship between input variables defining an output variable. For the convenience of presentation, the dots are called control points.



4.4.a 4.4.b 4.4.c

Figure 4.4. Control surfaces with different physical value domains.

As pointed out, the physical value domain of the input and output variables and the fuzzy rule base system are the main factors that determine the performance and quality of the fuzzy controller. There is no general method for selecting and adjusting the physical value domains and establishing the fuzzy rule base system, which is mainly based on expert knowledge and experiments. However, some problems can be the following approach:

The estimated physical value domains can hardly coincide with the actual value domain, the deviation is inevitable, but the deviation cannot be too large, and the selection of the physical value domain is based on the following analysis:

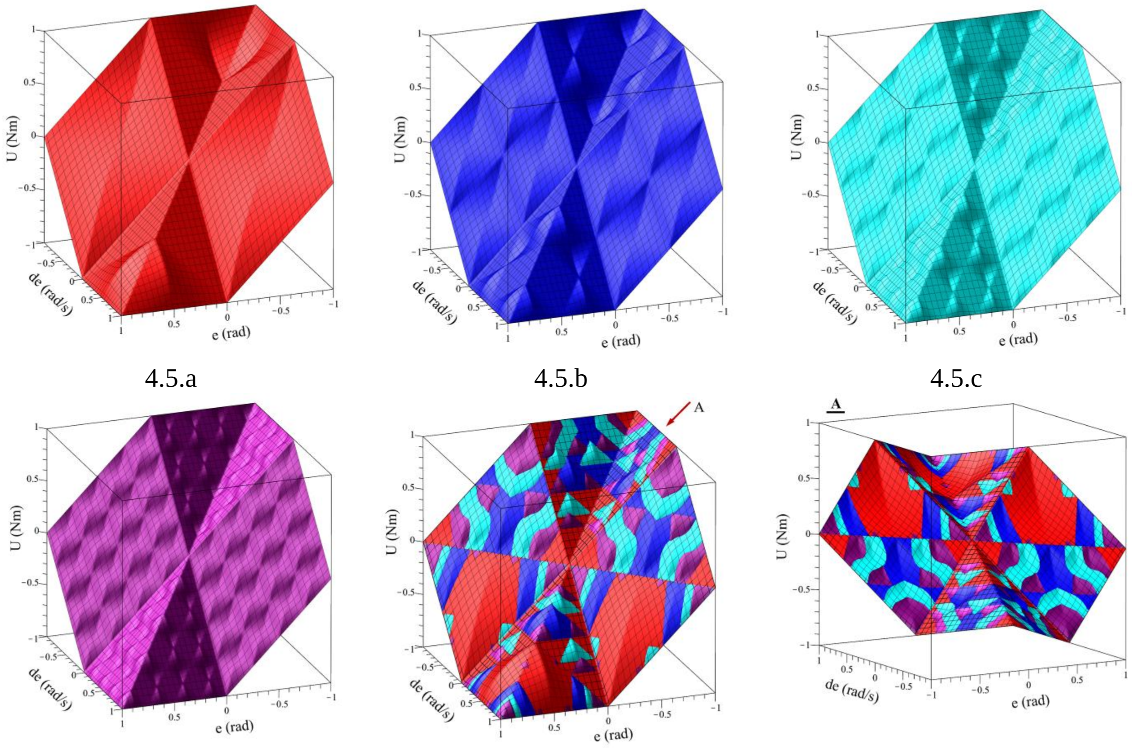
The input signal is usually the deviation between the desired value and the actual value of the state variable, so it is fine when the values of the input signals are small. Therefore, if the physical value domain of the input variables is selected as too large, it will be unreasonable.

However, if the physical value domain is too small, there will be cases where many actual values of the input signal are outside the estimated value range, then the value of the membership function is assigned to 0 or 1 by default. This leads to the slow tuning of the output signal. This explains that when there is a deviation of the motion trajectory, the system's return to the correct state by the fuzzy controller may be slower than by the crisp controller (see also in section 5).

The appropriate physical value domain would be such that the control points are located near or mainly concentrated in its center, i.e., in the central region of the control surface, as shown in Figure 4.4.c.

Another problem when designing fuzzy controllers is the number of quantization levels of universe sets of language variables. Figures 4.5.a - 4.5.d show control surfaces corresponding to 3, 5, 7, 9 fuzzy sets created from partitioning universe sets. Figures 4.5.e and 4.5.f show all four control surfaces simultaneously for easy observation and comparison. It can be seen that, with the function of triangle and trapezoid, the difference between the control surfaces is not too significant, Figures 4.5.a - 4.5.f. At the quantization level of 3, the control surface is simpler and coarser, but the computational volume will be less.

In contrast to the partition level of 9, the control surface is smoother, and the computational volume is more significant. However, the difference between the control surfaces is not too substantial, reflecting the numerical experiment that the degree of accuracy improvement by increasing the quantization level is not too great. Therefore, partition levels 3, 5, 7, 9 [17, 18, 26, 27, 51, 57, 58, 62,...] are still commonly used depending on the control object, but the most common is at the quantization level of 5 and 7 fuzzy sets.



4.5.a 4.5.b 4.5.c

4.5.d 4.5.e 4.5.f

Figure 4.5. Control surfaces with different quantization levels of the discourse universe of linguistic variables.

4.4. Defuzzification

As pointed out in published works, many methods for defuzzification techniques have been studied and applied. Among them, the most popular and effective is the Center Of Area method (COA) [17, 18, 26, 27, 57, 58, 62].

$$u' = \frac{\int_{S'} u\mu_{F'_u}(u)\,du}{\int_{S'} \mu_{F'_u}(u)\,du}; \quad \text{or} \quad u' = \frac{\sum_{j=1}^{N} u_j \mu_{F'_{ui}}(u)}{\sum_{j=1}^{N} \mu_{F'_{ui}}(u)} \tag{4.8}$$

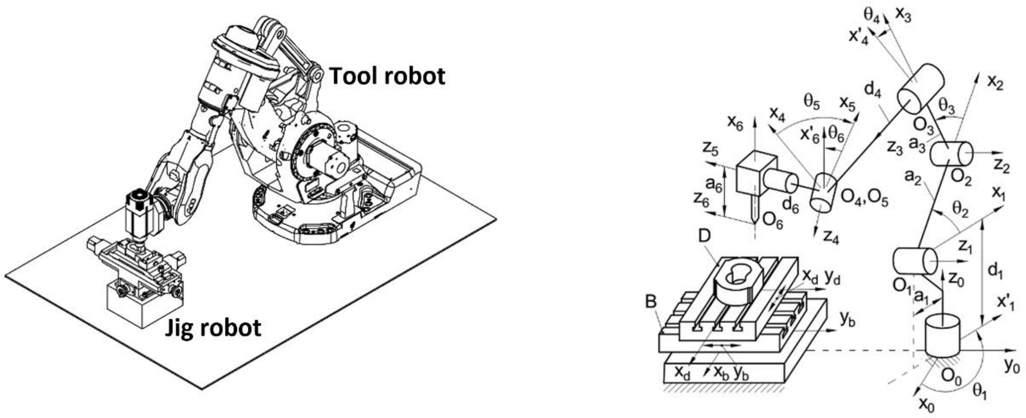
The first formula is for the continuous u case; u′ is the actual output value, taken according to the coordinates of the region's centroid, formed by the membership function line and the horizontal axis; S′ is the domain of the fuzzy set F'_u.

5. ROBOT FUZZY CONTROLLER IN MECHANICAL ENGINEERING

Based on research results on controlling manipulators in general as well as mechanical machining robots [27, 57, 62], this section analyzes the design of fuzzy controllers for mechanical machining robots, focusing on the following problems: The main topics mentioned are fuzzy set partitioning, determining the physical value domain of input and output variables, creating fuzzy rule base system and inference mechanism. The problems are analyzed based on the application of the robot model with numerical simulation experimental results.

As pointed out above, technology-manipulating robots in general and mechanical processing robots often have a manipulator-like structure with two typical kinematic structures: open chain and closed chain.

5.1. Fuzzy Controller for Open Chain Cooperative Robot



5.1.a 5.1.b

Figure 5.1. Model of a collaborative robot system, mechanism of relative manipulation, with 8 degrees of freedom for milling – MRM8.

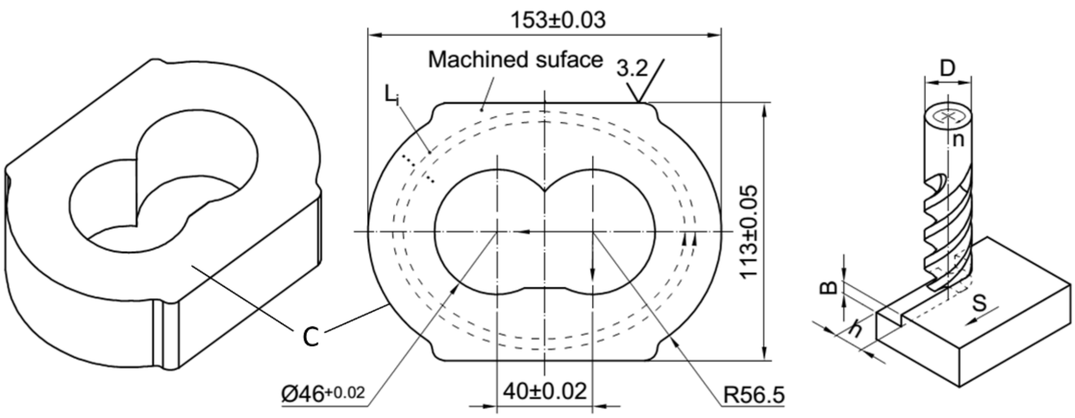


Figure 5.2. Machining details and toolpaths.

Figure 5.1 shows a system model of two mechanical machining manipulators, including: (1) a manipulator gripping and moving the machining tool, from now on referred to as a tool robot, and (2) a robot with table form that clamps and moves the workpiece, from now on referred to as table robot or jig robot (Fig. 5.1.a). These collaborative robots perform coordinated manipulations through the relative motion of two manipulator mechanisms. Therefore, this kind of collaborative robot system is also called the Mechanism of Relative Manipulation - MRM [93-95]. Figure 5.1.b is a diagram of the kinematic structure representing the motion of a tool robot consisting of eight degrees of freedom of motion, in which 6 degrees of freedom of the tool robot are represented by the joint coordinates θ1, …, θ6, and two degrees of freedom of movement of the jig robot, represented by yb, xd. The MRM8 notation for robots with eight degrees of freedom is considered here.

Figure 5.2 shows the workpiece with the surface to be machined as C. The machining process is performed by the coordinated motion between the tool robot and the jig robot so that the tool moves along the tool paths Li, with the position and orientation of the cutting tool to ensure a flat surface cut C. From the requirement for such manipulative motion, the motion of the links at the joints is calculated in terms of position and velocity over time and is input data to the controller.

In the case of a crisp controller, the control forces are calculated based on the dynamic model and according to the position and velocity input at the joints. It is difficult to accurately determine the quantities in the dynamic model of such a system, especially since the cutting force at the contact position between the cutter and the workpiece is an uncertain, ever-changing factor.

Depending on the capabilities created by design, the fuzzy controller can calculate and determine some parameters of the dynamic model, calculate error compensation, completely replace the dynamic model, and calculate the control force without needing a dynamic model. In the works [27, 57, 62], the fuzzy controller does not use the dynamic model when calculating the driving force, only relying on the input signal about the position error and the velocity error between the desired value and the actual value.

The contents and methods of implementing the design steps are presented in Section 4. Here only a few key issues are further analyzed below.

Input and output language variables

The surveyed robotic machining system has 8 degrees of freedom. The input variables will be eight signal pairs, each corresponding to each joint, including the position and velocity errors between the desired and actual values. Each joint has an output language variable which is the control force. For simplicity, the above linguistic variables can be abbreviated as the linguistic variables of position, velocity, and control force. The language variables of position $e_i(t)$, velocity $\dot{e}_i(t)$, and control force $U_i(t)$. i = 1,..,n is denoted as below:

$$\begin{cases} e_{(n\times1)} = \left[e_1, e_2, ..., e_n\right]^T; \quad e_i = q_i - q_{di}; \quad i = 1,...,n \\ \dot{e}_{(n\times1)} = \left[\dot{e}_1, \dot{e}_2, ..., \dot{e}_n\right]^T; \quad \dot{e}_i = \dot{q}_i - \dot{q}_{di}; \quad i = 1,...,n \end{cases} \tag{5.1}$$

$$U = \left[U_1, U_2, ..., U_n\right]^T; \quad i = 1,...,n \tag{5.2}$$

where q_i –actual value, and q_{di} - the desired value of the position variable; $\dot{q}$, $\dot{q}_{di}$ are the actual and desired values of the velocities of the joints, respectively; i=1,..,n=8.

Partition of discursive universes for input and output variables

As analyzed in Section 4, linguistic variables' discursive universe quantization levels can usually be from 3, .., 9, but the common ones are 5, 7. The works [26, 27, 51, 57, 62] have applied a quantization level of 5, achieving high accuracy while computational volume is reasonable. Thus, for each pair of input signals and one output signal for each joint, the universe of discourse for linguistic variables of position, velocity, and control force is partitioned into a partitioned discourse universe, and determine the number of fuzzy sets and the corresponding number of linguistic terms, as shown in Table 4.1.

In general, universes of discourse can be arbitrarily partitioned with fuzzy sets defined on them of the corresponding linguistic variables. The input data simultaneously affects the output data. On the other hand, in terms of operating principle, the links and joints of the robot have the same dynamic characteristics. Therefore, to simplify while ensuring controller efficiency, the universe set of linguistic variables is partitioned to the same number of levels and with a uniform proportional mapping. Therefore, the fuzzy sets and linguistic terms are denoted the same for the linguistic variables corresponding to the joints. Although the symbols are the same, the physical values of language variables are determined by their respective physical value domains, as shown in (4.1).

Determine the membership function of language variables

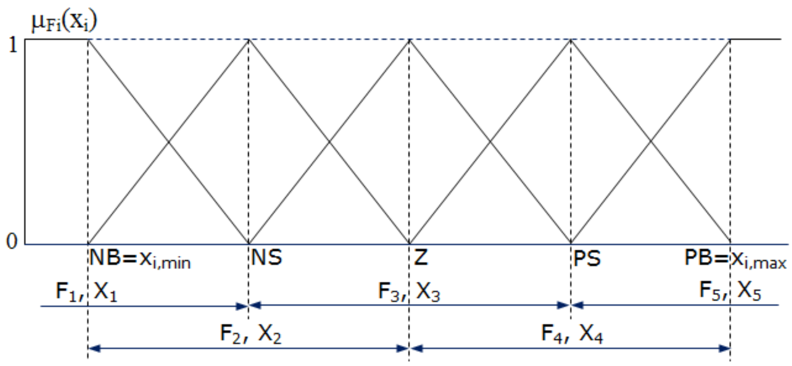


Figure 5.3. Triangle and trapezoidal membership functions.

As is known, many types of membership functions can be used, but the triangle and trapezoidal functions are the most commonly used. The reason for this may be the easy and simple calculation and the fit of the system's dynamic characteristics to this form of membership function. This is shown in the works [26, 27, 51, 57, 62], where the membership functions of the left trapezoid, the triangle, and the right trapezoid are used to represent the member values of the language variables e(t), $\dot{e}(t)$ and U(t), Figure 5.3. Although the dynamic system is complex, the fuzzy controller in [26, 27, 51, 57, 62] designed with these membership functions still achieves high accuracy.

Fuzzy Composition Device

Creating a fuzzy rule base system and inference mechanism is applied according to the method given in Section 4. Creating a fuzzy rule of the form (4.2) is firstly based on knowledge of the system's dynamics. Let's analyze a rule of (4.2) as follows:

$$\text{If } e(t) \text{ is Nagative Big and } \dot{e}(t) \text{ is Nagative Big then } u(t) \text{ is Positive Big} \qquad (4.2)$$
...

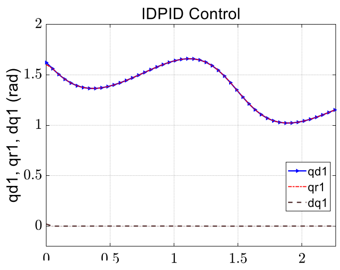
From the identification of signals e(t), according to (5.1), rule (4.2) indicates that deviations with large values of position and velocity (Negative Big) need to increase control force (Positive Big) so that the actual position and velocity progress to the desired position and velocity. This rule would need no proof, and it is all the more obvious when analyzing the law as Newton's 2nd axiom that acceleration is proportional to the force applied. Thus, with this rule, the membership functions of the triangle and trapezoid shapes are a suitable choice. The inference mechanism and the method of aggregation of appropriate rules from this obvious rule-based system will create a controller with the desired performance and accuracy.

Based on the publications in [57], below will analyze the method of correcting the physical value domain and the rule-based system based on numerical simulation experiments.

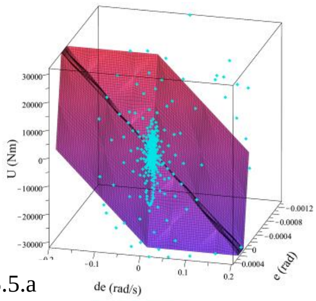
Comparing the desired trajectory with the actual trajectory obtained from the fuzzy controller operation allows for confirming the reliability of the fuzzy controller. However, here the control results of a crisp controller are also quoted for further comparison. Figure 5.4 shows the controller's result with two feedback loops, including Inverse Dynamics + Proportional Integral Derivative - IDPID. The IDPID controller is assumed to have an absolutely accurately defined dynamic model, so the control results are accurate and a reliable basis for comparison.

Sub-figures 5.4.a,..,h represent the desired position, the actual position obtained from the controller operation, and the difference between them. For example, Figure 5.4.a represents qd1, qr1, and dq1 of joint 1.

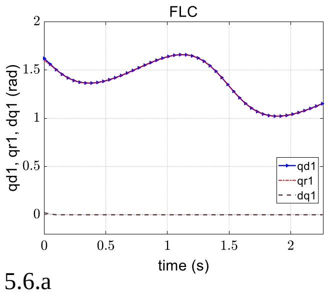
Figures 5.5 and 5.6 show correspondingly the control surfaces and simulation results of the fuzzy controller for the robot. The figures' grid surfaces in Figures 5.5.a,..,h are control surfaces represented based on the physical value domains, the fuzzy rule base system (4.2), and the rule inference and synthesis mechanism, as shown in Section 4. The control points (dots) in Figures 5.5.a,..,h are control points obtained from the control results (numerical simulation experiments) and are called real control points.



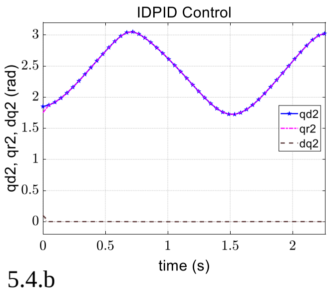
5.4.a



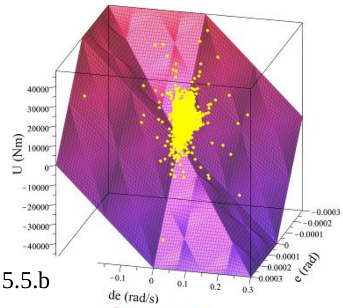
5.5.a



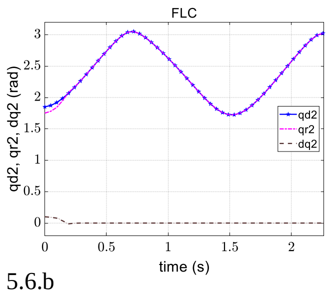
5.6.a



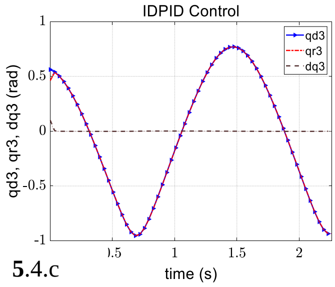
5.4.b



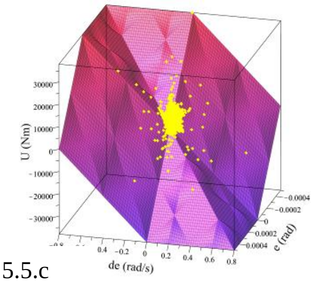
5.5.b



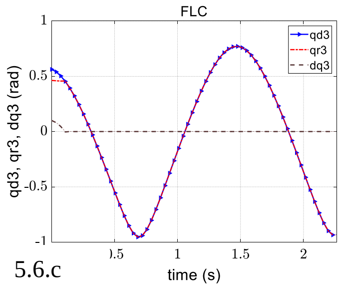
5.6.b



5.4.c



5.5.c



5.6.c

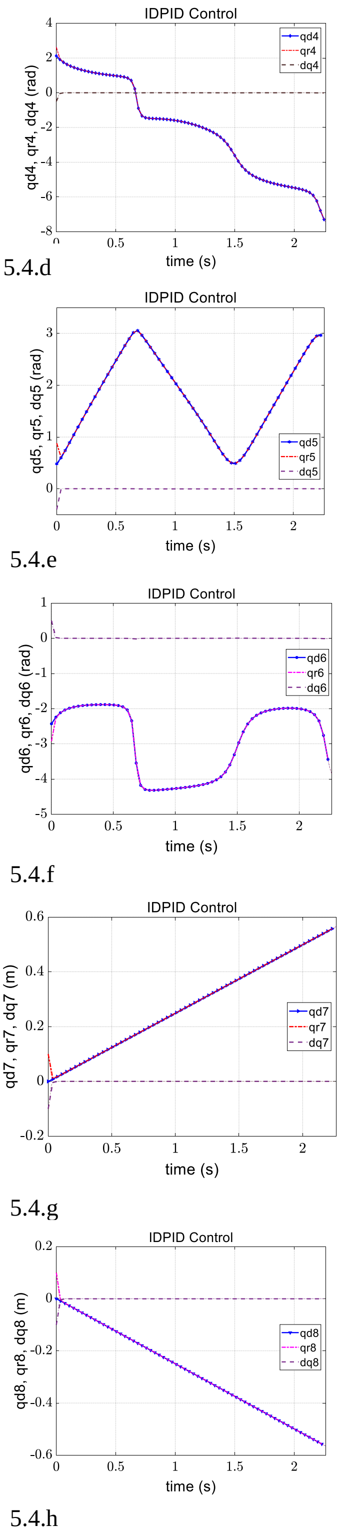


Figure 5.4. Control results of the IDPID controller.

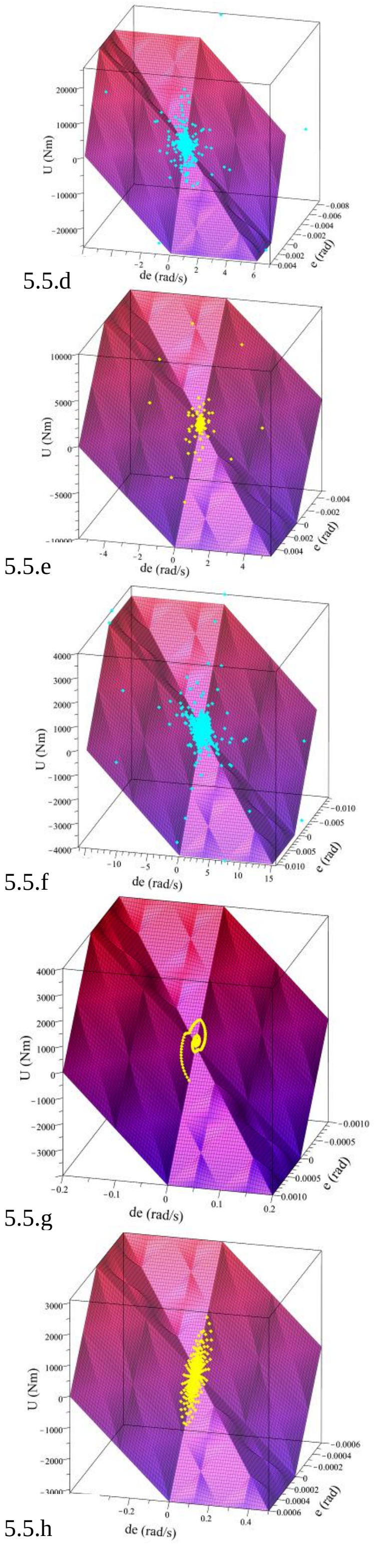


Figure 5.5. Design control Surf. and actual control points.

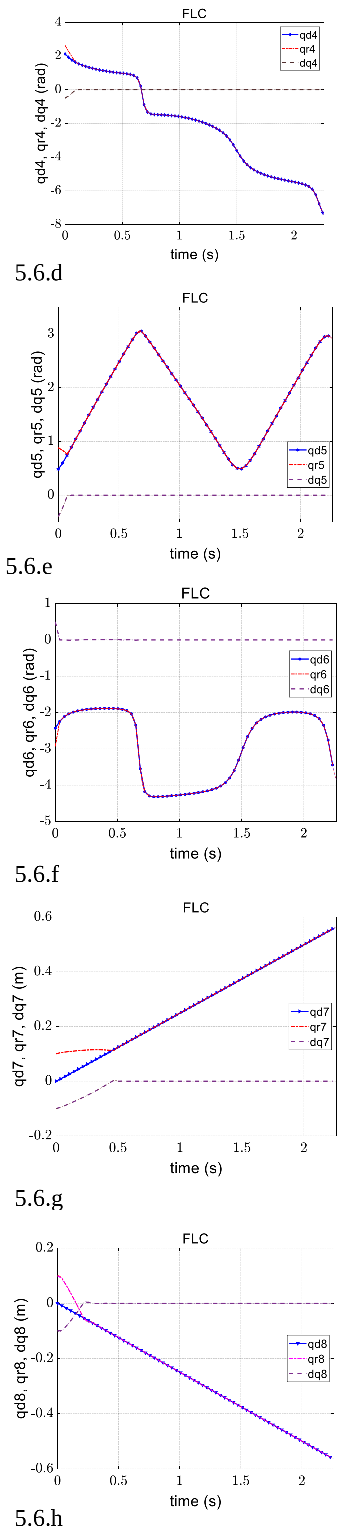


Figure 5.6. Control results of the Fuzzy controller.

If all real control points lie on the control surfaces, Figures 5.5.b,c,e,g,h, then it shows that the controller is operating efficiently. The physical value domain is estimated, and the established rule base system is appropriate. In fact, due to disturbances or deviations from the system's trajectory at a start or unusual time, some actual control points may be outside the control surface, Figures 5.5.a,d,f; but the fuzzy controller can still bring the system to trajectory. In such cases, points outside the designed control surfaces may be due to initial deviation or abnormal noise. The fuzzy controller will correct the deviation, and the control points will be concentrated mainly on and near the center of the control surfaces.

Figures 5.6.a,..,h show the control results for the motion of the joints. The symbols qr1, qr2, ..., qd1, qd2, ..., dq1, dq2, ..., represent the joint's actual position, desired position, and position error, respectively. This result shows that the fuzzy controller has good quality and high accuracy, and the actual and desired trajectory coincides.

Table 5.1 shows the physical value domain of the input and output variables e(t), de(t), and U(t) after some adjustments to get the control results, as shown in Figures 5.6.a,..,h.

Table 5.1. The physical value domain of language variables applied to robot controllers MRM8.

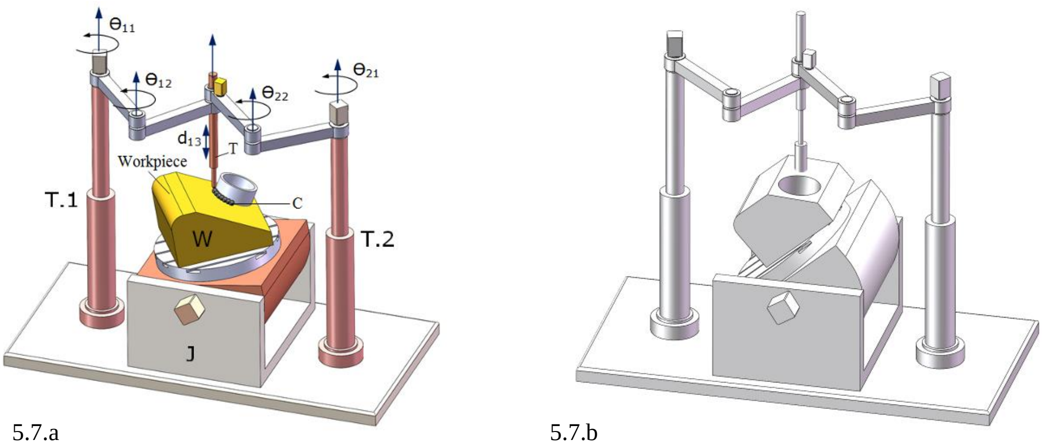
Joint	e_i	$\dot{e}_i$	u_i
1	[-0.0002, 0.0002] (rad)	[-0.2, 0.2] (rad/s)	[-32000, 32000] (Nm)
2	[-0.0003, 0.0003] (rad)	[-0.3, 0.3] (rad/s)	[-48000, 48000] (Nm)
3	[-0.0005, 0.0005] (rad)	[-0.8, 0.8] (rad/s)	[-38000, 38000] (Nm)
4	[-0.0037, 0.0037] (rad)	[-6.0, 6.0] (rad/s)	[-25500, 25500] (Nm)
5	[-0.0050, 0.0050] (rad)	[-5.5, 5.5] (rad/s)	[-10000, 10000] (Nm)
6	[-0.0060, 0.0060] (rad)	[-10.5, 10.5] (rad/s)	[-4000, 4000] (Nm)
7	[-0.0010, 0.0010] (m)	[-0.2, 0.21] (m/s)	[-4000, 4000] (N)
8	[-0.0006, 0.0006] (m)	[-0.5, 0.5] (m/s)	[-3100, 3100] (N)

It is found that the results of the fuzzy controller (Figure 5.6) are as good as the results of the crisp controller (Figure 5.4). Observing the graphs at the beginning of the control process; there is a deviation between the desired and the actual trajectory, and the fuzzy controller brings the system to the correct trajectory a little slower than the controller. For clarity, this has been discussed in Section 4.3.3.

Based on the knowledge of robot dynamics and observation in Figure 5.5.a,..,h, and comparising between the designed control surface and the response part, which are real control points, the physical value domain of linguistic variables can be adjusted as well as correction of fuzzy rules. This method is easy to apply when performing numerical simulation experiments as well as adjusting robot control in actual installations.

5.2. Fuzzy Controller for Closed Chain Cooperative Robot

The manipulation robot system consists of two manipulators that coordinate technological manipulation, the first manipulator carrying a machining tool (Tool Robot - T) with three degrees of freedom. The second movable table-type manipulator has two degrees of freedom, acting as a jig for clamping and moving the workpiece, from now on referred to as the table robot or the Jig Robot (Jig Robot - J). The robotic system has five degrees of freedom of movement (so-called Mechanism of Relative Manipulation-MRM5). The workpiece (Workpiece – W) is positioned and clamped on the J table robot. Usually, to process the machinery detail with any geometric shape and surface position, the robot needs many degrees of freedom due to the need to perform complex forming machining movements. With a cylindrical and conical machining surface, as shown in Figure 5.7, the machined surface's position is arbitrary for the positioning standard of the workpiece; the five degrees of freedom robot, as stated above, meet the machining requirements.



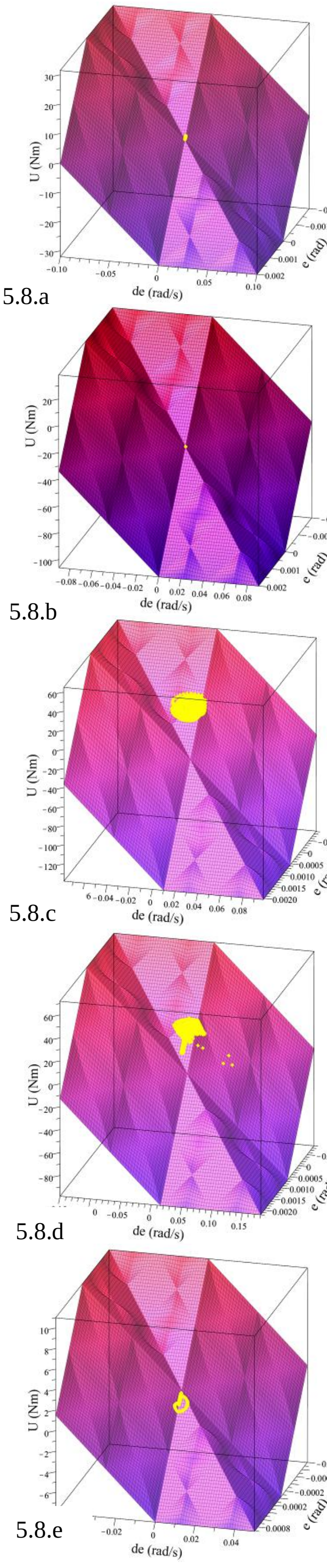
5.7.a 5.7.b

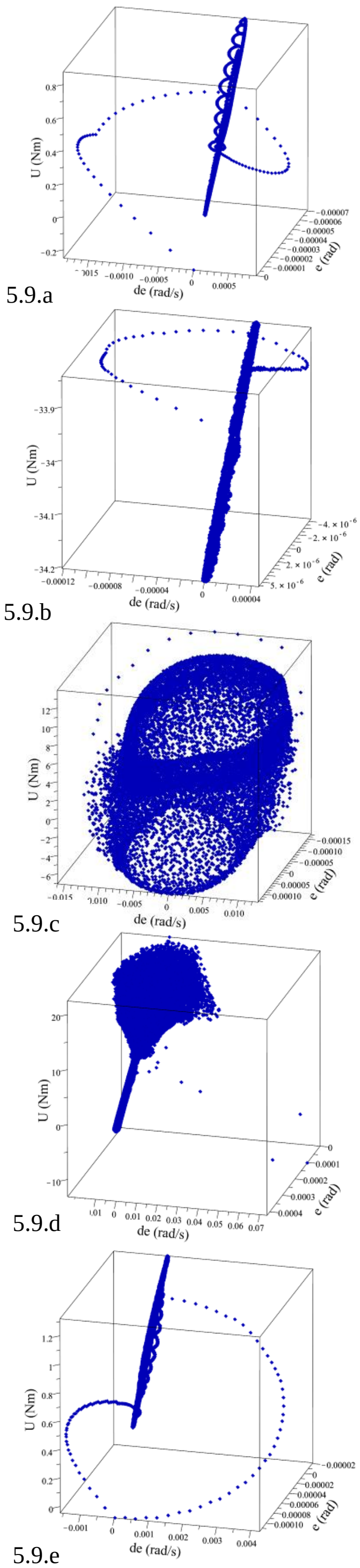
Figure 5.7. Collaborative robot closed chain structure with mechanism of relative manipulation, five degree of freedom (dof) – MRM5.

The tool robot has a closed chain structure to ensure rigidity. As such, this robot is close to a 5-axis CNC machine. However, thanks to this structure, the robot can be designed with a size that meets the machining requirements, i.e., the robot has a larger workspace than a CNC machine. On the other hand, thanks to the flexible programming ability, the robot's processing ability is flexible. Figure 5.7.a shows a robot that can perform arc welding to join machinery detail, and Figure 5.7.b shows a robot that can perform cutting milling or laser machining, etc.

The crisp controller for the closed-chain robot should be based on the dynamic model according to (3.7) - (3.17); different from the open-chain robot control, the dynamic model is (3.6) - (3.13). Calculating the dynamic model (3.7) - (3.17) will be more complex and complicated. When applying the fuzzy control method with different types of robots, it can be based on the general process as described in Section 4 and Section 5.1. In [62] we presented the controller for the arc welding robot, the steps of designing the fuzzy controller, estimating and calibrating the physical value domain, creating a rule base system and an inference mechanism, and defuzzification, similarly performed as with the robot described in Section 5.1. The tool robot that clamps and moves the welding head coordinates the movement, with the jig robot that clamps and moves the workpiece. The motion law of the links at the joints of the robot is designed and is the input data of the controller. The fuzzy controller in [62] is designed so that the control force (output variable) can be calculated to ensure the robot's motion without relying on the dynamic model.

Figures 5.8.a,..,e show control surfaces and control points, in which the control surfaces are determined from the selection of the physical value domain, the fuzzy rule base system and the fuzzy inference mechanism are designed. The control points (dots) located on the control surface are obtained from the numerical experimental results of the controller simulation.





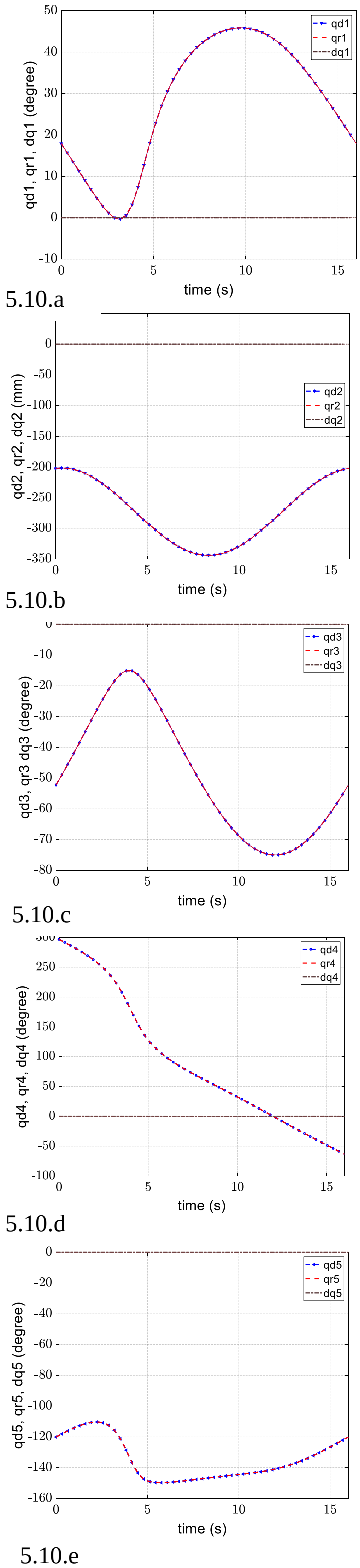


Figure 5.8. Design control Surf. and actual control points. *Figure 5.9.* Real control points. *Figure 5.10.* Control results.

Thanks to some corrections to the physical value domain and the rule-based system based on numerical simulation experiments, the control points are all located on the control surfaces. They are primarily concentrated in the center of the control surfaces. To observe more clearly, Figure 5.9.a,..,e shows the control points. With centralized control points, as shown in Figures 5.8.a,..,e, it shows that the controller works effectively with high accuracy, as shown in Figure 5.10.a,..,e.

Figures 5.10.a,..,e show the control results for the motion of the joints. The symbols qr1, qr2, ..., qd1, qd2, ..., dq1, dq2, ..., represent the joint's actual position, desired position, and position error, respectively. This result shows that the fuzzy controller has good quality and high accuracy, and the actual and desired trajectory coincides.

Table 5.2 presents the physical value domains determined after some adjustment steps to achieve control results, as shown in Figures 5.10.a,..,e.

For simplicity and ease of application, in this paper, a method of designing fuzzy controllers for mechanical machining robots is presented based on the steps of selection and calibration, as shown in Section 4, and Sections 5.1 and 5.2. These manual methods are simple and still achieve results suitable for complex systems such as machining robots.

Table 5.2. The physical value domain of language variables applied to robot controllers MRM5.

Joint	e_i	$\dot{e}_i$	u_i
1	[-0.002, 0.002] (rad)	[-0.1, 0.1] (rad/s)	[-32, 31.5] (N.m)
2	[-0.002, 0.002] (rad)	[-0.09, 0.09] (rad/s)	[-106, 38] (N.m)
3	[-0.00092, 0.00205] (m)	[-0.078, 0.098] (m/s)	[-138, 65] (N)
4	[-0.0009, 0.0022] (rad)	[-0.15, 0.18] (rad/s)	[-98, 72] (N.m)
5	[-0.001, 0.00092] (rad)	[-0.05, 0.05] (rad/s)	[-8.0, 11] (N.m)

However, to achieve better results, intelligent algorithms can be applied. For example, genetic algorithms and modern evolutionary algorithms can be used to determine the physical value domain of input and output variables. Algorithms based on artificial intelligence, deep learning, reinforcement learning, and deep reinforcement learning can be integrated and used in training and correcting the rule base system, etc. [55, 56].

5.3. Stability of the Fuzzy Controller

There have been many research and analysis works on the stability of fuzzy controllers [18, –100-109], but this is still an open problem. As mentioned, fuzzy sets, fuzzy logic theory, and fuzzy controllers generally work on uncertain and "vague" information. Therefore, proving the stability according to the method based on the mathematical model as with the crisp controller is challenging.

Does the proof, as with the crisp controller will, lose the "fuzzy" and reduce the flexibility of the fuzzy logic-based working system? Here, starting from working with vague, uncertain information but with reasonable human-like inference, we briefly discuss the stability of the fuzzy controller.

Let's take an example. The dancer holds a stick upright on the tip of one finger, which at the top of the stick has an egg. A not-so-excellent dancer can keep the stick and the egg from falling with just a simple operation. That's when the stick tends to lean in that direction that the dancer moves his finger and the stick in that direction.

The fuzzy rule base system (4.2) is also built from simple, obvious things like natural human inference. From the rule system (4.2) and considering the control surfaces (Figures 4.3-4.5, 5.5, 5.8) represented from the rule system, it can be seen that when the error increases, the control force decreases. On the other hand, for triangular and trapezoidal membership functions, the change of the control force is linear and bounded by the physical value domain (see Figures 4.3-4.5, 5.5, 5.8). Thus, the error will decrease with the operation of the fuzzy controller. This is reasonable and consistent with the ambiguous nature of the fuzzy logic-based working system.

6. CONCLUSION

The article has introduced the advantages of applying robots in mechanical processing, an inevitable development trend. Controlling the robot to ensure the requirements of mechanical processing techniques and efficiency is still an open problem. From there, the article analyzes and presents the main problems in the design process of fuzzy controllers for mechanical machining robots.

The method of analyzing, selecting, and correcting the physical value domain, creating a fuzzy rule base system and the inference mechanism, synthesizing the rule system, and the defuzzification method shown in Sections 4 and 5 is simple and effective. Although there are fuzzy inference methods and fuzzy composition rules synthesis such as those of Mamdani and Takagi-Sugeno,..., the paper focuses on one of Mamdani's methods, which is easy to apply for newbies working with fuzzy controllers and, at the same time, gives good results.

As mentioned above, setting up and calibrating the controller can be done simply by hand, based on numerical simulation experiments presented in Sections 4 and 5. This is easy for beginners to use fuzzy logic to design fuzzy controllers.

The next development step is to apply intelligent algorithms such as genetic algorithms, modern evolutionary algorithms, artificial intelligence, deep learning, reinforcement learning, deep reinforcement learning, etc. to identify the physical value domain and create the appropriate rule system.

The problems mentioned in the article are undoubtedly helpful in applying fuzzy logic to control problems in general and mechanical processing robots. A good application of these basics and simplicity is also the basis for developing studies with Type-2 Fuzzy, integrating Adaptive Fuzzy Controller, Sliding Fuzzy Controller, and other intelligent algorithms, such as the Hedge Algebraic Fuzzy Ccontroller, Neural Fuzzy Controller, etc. with modern evolutionary algorithms.

Declaration of competing interest. The author declares that no competing financial interests or personal relationships could affect the work reported in this article.

Conflict statement: The author declares no potential conflicts of interest with respect to the research, authorship, and/or publication of this article.

REFERENCES

1. Iglesias I., Sebastián M. A., and Ares J. E. - Overview of the state of robotic machining: Current situation and future potential. Procedia engineering **132** (2015) 911-917. https://doi.org/10.1016/j.proeng.2015.12.577

2. COMET, E. U. FP7. - project: Plug-and-produce COmponents and METhods for adaptive control of industrial robots enabling cost effective, high precision manufacturing in factories of the future. (2011). http://www.cometproject.eu
3. Petko M., Karpiel G., Gac K., Góra G., Kobus K., Ochoński J. - Trajectory tracking controller of the hybrid robot for milling. Mechatronics **37** (2016) 100-111. https://doi.org/10.1016/j.mechatronics.2016.03.012
4. Gołda G., Kampa A. - Modelling of cutting force and robot load during machining. Advanced Materials Research. Trans Tech Publications **1036** (2014) 715-720. https://doi.org/10.4028/www.scientific.net/AMR.1036.715
5. Cen L., Melkote S. N. - Effect of robot dynamics on the machining forces in robotic milling. Procedia Manufacturing, **10** (2017) 486-496. https://doi.org/10.1016/j.promfg.2017.07.034
6. Kai Wu, Jiaquan Li, Huan Zhao and Yong Zhong - Review of Industrial Robot Stiffness Identification and Modelling. Applied Scince, **12**(17) (2022) 8719 1-24. https://doi.org/10.3390/app12178719
7. Pandremenos J., Christos D., Panagiotis S., George C. - Machining with robots: a critical review. Proceedings of DET2011 (2011) 1-9. https://www.researchgate.net/publication/265552374
8. Ji, Wei, and Lihui Wang. - Industrial robotic machining: a review. The International Journal of Advanced Manufacturing Technology **103.1** (2019) 1239-1255. https://doi.org/10.1007/s00170-019-03403-z
9. Denkena, Berend, Benjamin Bergmann, and Thomas Lepper - Design and optimization of a machining robot. Procedia Manufacturing **14** (2017) 89-96. https://doi.org/10.1016/j.promfg.2017.11.010
10. Lukasiewicz J. O logice trojwartosciowej. Ruch Filozoficzny, (1920) 5, 170-171. English translation, On three-valued logic, in "Jan Lukasiewicz - Selected Works" (Borkowski L. editor), North-Holland Publishing Company, Amsterdam, (1970) 87-88. https://www.scribd.com/document/359602256/J-Lukasiewicz-Selected-Works-L-Borkowski-Editor#
11. Lukasiewicz J., Tarski A. Investigation into the sentential calculus, in Alfred Tarski, "Logic, Semantics, Metamathematics", papers from 1923 to 1938. Clarendon Press (1956). https://www.amazon.com/Logic-Semantics-Metamathematics-Papers-1923/dp/0915144751
12. Zadeh L. A. - Fuzzy sets. Information and control **8** (1965) 338-353. https://doi.org/10.2307/2272014
13. Zadeh L. A. - Fuzzy logic and approximate reasoning. Synthese **30** (1975) 407-428. https://doi.org/10.1007/BF00485052
14. Mamdani E. H. - Application of fuzzy algorithms for control of simple dynamic plant. Proc. IEEE **121** (12) (1974) 1585-1588.
15. Mamdani E. H. - Twenty years of fuzzy control: experiences gained and lessons learnt. In Proceedings Second IEEE International Conference on Fuzzy Systems (1993) 339-344. https://ieeexplore.ieee.org/document/327439
16. George J. Klir, Bo Yuan. - Fuzzy sets and Fuzzy Logic. Prentice Hall PTR (1995).
17. Lee K. H. - First course on fuzzy theory and applications. Springer Science & Business Media (2004).

18. Hooda D. S., Raich V. - Fuzzy logic models and fuzzy control: An introduction. Alpha Science International (2016).
19. Hampel R., Wagenknecht M., Chaker N., (Eds.). - Fuzzy control: Theory and practice. Springer Science & Business Media (2013).
20. De Silva C. W. - Intelligent control: fuzzy logic applications. CRC press (2018).
21. Terrell Harvey, Dallas Mulling - Fuzzy modeling and control methods, applications and research. NOVA (2018).
22. Timothy J. Ross. - Fuzzy logic with engineering applications. Willey 4rd Edition (2017).
23. Algorithm with Type-2 Fuzzy Logic for Dynamic Parameter Adaptation with Application to Intelligent Control. Springer (2021).
24. Ardashir M., Mohammad H. S., Chunwei Z., Oscar C., Rathinasamy S., Fayez F. M. El-Sousy. - Modern Adaptive Fuzzy Control Systems. Springer (2022). https://link.springer.com/book/10.1007/978-3-031-17393-6
25. Alexander C. A. Dino D. L., Elmer P. D., Levin J. S. C., Michael C. D. P. Del Rosario, Jho N. S. K. - Fuzzy logic-controlled 6-DOF robotic arm color-based sorter with machine vision feedback. International Journal of Advanced Computer Science and Applications **9** (5) (2018) 20-31. Doi: 10.14569/IJACSA.2018.090503
26. Toan N. V., Khoi, P. B. - Fuzzy-based-admittance controller for safe natural human–robot interaction. Advanced Robotics **33** (2019) 1-9. https://doi.org/10.1080/01691864.2019.1607551
27. Khoi P. B., and Hong N. X. - Fuzzy Logic-Based Controller for Bipedal Robot. Applied Sciences **11** (24) (2021) 11945 1-28. https://doi.org/ 10.3390/app112411945
28. Aminurrashid N., Azhan A. R., Nur A. I., Mohd A. S., Adzni M. S., Nor A. M., Nurfaizey A. H., Faizil W., Mohd Z. A. and Mustafa S. K. - A Performance Comparison of PID and Fuzzy Logic Control Methods for Trajectory Tracking of Wheeled Mobile Robot. International Journal of Nanoelectronics and Materials. Special Issue,Vol. **14** August (2021) 387-394. http://dspace.unimap.edu.my:80/xmlui/handle/123456789/73365
29. Medjoubi, Halima, Abdessemad Yassine, and Hassam Abdelouahab - Design and Study of an Adaptive Fuzzy Logic-Based Controller for Wheeled Mobile Robots Implemented in the Leader-Follower Formation Approach. Engineering, Technology & Applied Science Research **11.2** (2021) 6935-6942. https://doi.org/10.48084/etasr.3950
30. Sharma K. D., Chatterjee A., Rakshit A. - Intelligent Adaptive Fuzzy Control, In Intelligent Control. Springer, Singapore (2018) 3-21. https://doi.org/10.1007/978-981-13-1298-4_1
31. Do Khac Tiep, Kinam Lee, Dae-Yeong Im, Bongwoo Kwak, and Young-Jae Ryoo - Design of fuzzy-PID controller for path tracking of mobile robot with differential drive. International Journal of Fuzzy Logic and Intelligent Systems **18.3** (2018) 220-228. https://doi.org/10.5391/IJFIS.2018.18.3.220
32. Chiraz Ben Jabeur and Hassene Seddik - Design of a PID optimized neural networks and PD fuzzy logic controllers for a two-wheeled mobile robot. Asian Journal of Control **23.1** (2021) 23-41. https://doi.org/10.1002/asjc.2356
33. Kumar V., Nakra B. C., Mittal A. P. - A review on classical and fuzzy PID controllers. International Journal of Intelligent Control and Systems **16.3** (2011) 170-181.
34. Wolfer, James, and Chad A. George. - Fuzzy logic control for robot maze traversal: an undergraduate case study. World Congress on Computer Science, Engineering, and Technology Education (2006) 46-50.

35. Payman S., Olga D., Andrzej O., Gordana C. - Teaching fuzzy logic control based on a robotic implementation. IFAC Proceedings **46.17** (2013) 192-197. https://doi.org/10.3182/20130828-3-UK-2039.00047
36. Manuel A. M., Daniel R. A., Leonardo E. S., Nelson F. V.. - Fuzzy logic controller for cooperative mobile robotics implemented in leader-follower formation approach. Revista Facultad de Ingeniería Universidad de Antioquia, **76** (2015) 19-29. DOI: 10.17533/udea.redin.n76a03
37. Angga Rusdinar, and Sungshin Kim - Modeling of vision based robot formation control using fuzzy logic controller and extended Kalman filter. International Journal of Fuzzy Logic and Intelligent Systems **12.3** (2012) 238-244. https://doi.org/10.5391/IJFIS.2012.12.3.238
38. Jamaludin Jalani, and Suman Jayaraman - Design a fuzzy logic controller for a rotary flexible joint robotic arm, MATEC Web of Conferences. EDP Sciences **150 (**2018) 1-6. https://doi.org/10.1051/matecconf/201815001011
39. Ching H. C., Chien C. W., Yi T. W. and Po T. W. - Fuzzy logic controller design for intelligent robots. Mathematical Problems in Engineering 2017 (2017) 1-12. https://doi.org/ 10.1155/2017/8984713
40. Hua Zhang. - Adaptive Fuzzy Controller Design for Uncertain Robotic Manipulators Subject to Nonlinear Dead Zone Inputs. Computational Intelligence and Neuroscience, (2022) 9173249 1-8. https://doi.org/10.1155/2022/9173249
41. Pedro N., Nuno M., Norberto P. J. - CAD-based robot programming: The role of Fuzzy-PI force control in unstructured environments. 2010 IEEE International Conference on Automation Science and Engineering. IEEE (2010) 362-367.
42. Azura C. S., Erny A. A., Ribhan Z. A. R., Li H. F.. - Effect of fuzzy logic controller implementation on a digitally controlled robot movement. Kathmandu University Journal of Science, Engineering and Technology **4.1** (2008) 28-39. https://doi.org/10.3126/kuset.v4i1.2881
43. Katbab A. - Fuzzy logic and controller design-a review. Proceedings IEEE Southeastcon'95. Visualize the Future, IEEE (1995) 443-449. DOI: 10.1109/SECON.1995.513133
44. Touati Youcef, and Yacine Amirat - Fuzzy logic controller design methodology for Cartesian robot control. International journal of computer applications in technology **27.2** (2006) 85-96. https://doi.org/10.1504/IJCAT.2006.011135
45. Wakileh, B. A. M., Gill K. F. - Use of fuzzy logic in robotics. Computers in industry **10** (1988) 35-46.
46. Ade S. H., Nyayu L. H., Siti N., Irsyadi Y. - Application of type-1 and type-2 fuzzy logic controller for the real swarm robot, iJOE **15.6** (2019) 83-98. https://doi.org/10.3991/ijoe.v15i06.10075
47. Affane F., Meguenni K., and Omari A. - Type-2 fuzzy logic controller optimized by wavelet networks for mobile robot navigation, Indones. J. Electr. Eng. Comput. Sci. **18** (2020) 326-334. DOI: http://doi.org/10.11591/ijeecs.v18.i1.pp326-334
48. Chun T. C., Nana S., Juing S. C., and Chi J. W. - Equivalence between Fuzzy PID Controllers and Conventional PID Controllers, Appl. Sci. **7** (6) (2017) 513 1-12. https://doi.org/10.3390/app7060513

49. Juan M. B., David A. R., Sergio L., Juan José G., Donato H. and Diana M. P. - A Review on Applications of Fuzzy Logic Control for Refrigeration Systems, Applied Sciences **12.3** (2022) 1302. https://doi.org/10.3390/ app12031302
50. Khoi P. B.,Van Toan N. - Hedge-Algebras-Based Controller for Mechanisms of Relative Manipulation. International Journal of Precision Engineering and Manufacturing **19** (2018) 377-385. https://doi.org/10.1007/s12541-018-0045-8
51. Khoi P. B., Van Toan N. - Optimizing Neuro-Fuzzy System for Robot Control. Journal of Science and Technology **52** (2014) 685-699. https://doi.org/10.15625/0866-708X/52/6/3633
52. Kazuo Kiguchi and Toshio Fukuda. - Fuzzy neural controller for robot manipulator force control. Proceedings of 1995 IEEE International Conference on Fuzzy Systems (1995) 869-874. DOI: 10.1109/FUZZY.1995.409785
53. Oscar Castillo, Patricia Ochoa, Jose Soria - Differential Evolution Algorithm with Type-2 Fuzzy Logic for Dynamic Parameter Adaptation with Application to Intelligent Control. Springer (2021).
54. Alejandra Mancilla, Mario García-Valdez, Oscar Castillo and Juan Julian Merelo-Guervós - Optimal Fuzzy Controller Design for Autonomous Robot Path Tracking Using Population-Based Metaheuristics. Symmetry **14** (2) (2022) 1-17. https://doi.org/ 10.3390/sym14020202
55. Khoi Phan Bui, Giang Nguyen Truong, Dat Nguyen Ngoc **-** Modeling of Bipedal Locomotion by Combination of TD3 Algorithms and Graph Convolutional Network. Appl. Sci. **12(6)** (2022) 2948. 1-15. https://doi.org/10.3390/app12062948
56. Phan Bui Khoi, Nguyen Truong Giang, Hoang Van Tan - Control and Simulation of a 6-DOF BipedRobot based on Twin Delayed Deep Deterministic Policy Gradient Algorithm. Indian Journal of Science and Technology **14**(30) (2021) 2460-2471. doi.org/10.17485/IJST/v14i30.1030.
57. Khoi Bui Phan, Hai Thanh Ha, Sinh Vinh Hoang - Eliminating the effect of uncertainties of cutting forces by fuzzy controller for robots in milling process. Applied Sciences, **10(5)** (2020) 1685 1-31. https://doi.org/10.3390/app10051685
58. Khoi P. B., Van Toan N. - Application of fuzzy logic for controlling mechanisms of relative manipulation robot (MRM robot). J Sci Technol. **54** (2016) 385–401. https://doi.org/10.15625/0866-708X/54/3/7333
59. Liu H., Wang, T., Wang, D. - Constant cutting force control for CNC machining using dynamic characteristic-based fuzzy controller. Shock and Vibration (2015) 1-8. http://dx.doi.org/10.1155/2015/406294
60. Kovač P., Rodić D., Gostimirović M., Savković B., & Ješić D. - Application of adaptive neuro fuzzy systems for grinding process modeling. 6th International Conference on Information Society and Technology ICIST (2016) 247-250.
61. Huang S. J., Shy C. Y. - Fuzzy logic for constant force control of end milling. IEEE Transactions on Industrial Electronics **46** (1999) 169-176. DOI: 10.1109/41.744408
62. Van Toan N., Khoi P. B. - A control solution for closed-form mechanisms of relative manipulation based on fuzzy approach. International Journal of Advanced Robotic Systems **16** (2019) 1-11. https://doi.org/10.1177/1729881419839810
63. Hendzel Z., Burghardt A., Gierlak P., Szuster M. - Conventional and fuzzy force control in robotised machining. Solid State Phenomena. Trans Tech Publications **210** (2014) 178-185. DOI: 10.4028/www.scientific.net/SSP.210.178

64. Cuka B., Kim D. W. - Fuzzy logic based tool condition monitoring for end-milling. Robotics and Computer-Integrated Manufacturing **47** (2017) 22-36. https://doi.org/10.1016/j.rcim.2016.12.009
65. Hsu F. Y., Fu L. C. - Intelligent robot deburring using adaptive fuzzy hybrid position/force control. IEEE Transactions on Robotics and Automation **16** (2000) 325-335. DOI: 10.1109/70.864223
66. Hsu F. Y., Fu L. C. - A new adaptive fuzzy hybrid force/position control for intelligent robot deburring. In Proceedings 1999 IEEE International Conference on Robotics and Automation **3** (1999) 2476-2481. DOI: 10.1109/ROBOT.1999.770477
67. Basil M. A., Fernando M. and Agustın J. - Fuzzy controller for robot manipulators. International Fuzzy Systems Association World Congress. Springer, Berlin, Heidelberg (2007) 688-697.
68. Ahmed Kharidege, Ding Jianbiao, Yajun Zhang. - Performance Study of PID and Fuzzy Controllers for Position Control of 6 DOF arm Manipulator with Various Defuzzification Strategies. MATEC Web of Conferences **77** 01011 (2016) 1-6. DOI: 10.1051/matecconf/20167701011
69. Rezoug A., Boudoua S., and Hamerlain F. - Fuzzy logic control for manipulator robot actuated by pneumatic artificial muscles. Journal of electrical systems **1** (2009) 1-6.
70. Meng Bi - Control of Robot Arm Motion Using Trapezoid Fuzzy Two-Degree-of-Freedom PID Algorithm. Symmetry **12** (4) (2020) 665 1-10. https://doi.org/10.3390/sym12040665
71. Hua W., Yan X. W., Ling Z. M., Chang A. L. and Guo T. Y. - Hybrid velocity switching and fuzzy logic control scheme for cable tunnel inspection robot. Journal of Intelligent & Fuzzy Systems **29(6)** (2015) 2619–2627. DOI:10.3233/IFS-151965
72. Clarence W. de Silva - Applications of fuzzy logic in the control of robotic manipulators. Fuzzy Sets and Systems **70** (1995) 223 234. https://doi.org/10.1016/0165-0114(94)00219-W
73. Jakubiec Beata - Fuzzy logic controller for robot manipulator control system. Applications of Electromagnetics in Modern Techniques and Medicine. Proceedings IEEE (PTZE 2018) (2018) 77-80. DOI: 10.1109/PTZE.2018.8503205
74. Richa Sharma, Prerna Gaur, Mittalb A.P. - Design of two-layered fractional order fuzzy logic controllers applied to robotic manipulator with variable payload. Applied Soft Computing (2016) 1-12. https://doi.org/10.1016/j.asoc.2016.05.043
75. Seyed A. M., Mansour R., Ehsan Z., Amin Z. - Simulation and experimental control of a 3-RPR parallel robot using optimal fuzzy controller and fast on/off solenoid valves based on the PWM wave. ISA Transactions **61** (2016) 265-286. https://doi.org/10.1016/j.isatra.2015.12.005
76. Fahmy A. A., Abdel Ghany A. M. - Neuro-fuzzy inverse model control structure of robotic manipulators utilized for physiotherapy applications. Ain Shams Engineering Journal **4** (2013) 805-829. https://doi.org/10.1016/j.asej.2013.02.010
77. Richa Sharma, Rana K.P.S., Vineet Kumar - Performance analysis of fractional order fuzzy PID controllers applied to a robotic manipulator. Expert Systems with Applications **41** (2014) 4274-4289. http://dx.doi.org/10.1016/j.eswa.2013.12.030
78. Khalate, Amol A., Gopinathan Leena, and Goshaidas Ray - An adaptive fuzzy controller for trajectory tracking of robot manipulator, Intelligent Control and Automation **2.4** (2011) 364-370. DOI: 10.4236/ica.2011.24041

79. Abdul-Sadah, Ali Muhssen, Kamal MH Raheem, and Mohammed Mahdi Salih Altufaili - A fuzzy logic controller for a three links robotic manipulator, AIP Conference Proceedings **2386** No. 1 AIP Publishing LLC (2022). https://doi.org/ 10.1063/5.0066871
80. Hala Bezine, Nabil Derbel, Adel M. Alimi - Fuzzy control of robot manipulators: some issues on design and rule base size reduction. Engineering Applications of Artificial Intelligence **15** (2002) 401-416. https://doi.org/10.1016/S0952-1976(02)00075-1
81. Kuo-Ching Chiou, Shiuh-Jer Huang - An adaptive fuzzy controller for robot manipulators. Mechatronics **15** (2005) 151-177. https://doi.org/10.1016/ j.mechatronics.2004.07.005
82. Abraham M., Oscar C., Fevrier V., Jose S. and Mario G. - Optimal Design of the Fuzzy Navigation System for a Mobile Robot Using Evolutionary Algorithms. International Journal of Advanced Robotic Systems **10(2)** (2013) 1-11. https://doi.org/10.5772/55561
83. Rani Somiya, Amita Jain, and Oscar Castillo - Research trends on fuzzy logic controller for mobile robot navigation: A scientometric study, Journal of Automation, Mobile Robotics and Intelligent Systems (2019) 87-108. https://doi.org/10.14313/JAMRIS/1-2020/11
84. Saffiotti Alessandro, Enrique H. Ruspini, and Kurt Konolige - Using fuzzy logic for mobile robot control. Practical applications of fuzzy technologies. Springer, Boston, MA (1999) 185-205. https://doi.org/10.1007/978-1-4615-4601-6_5
85. Singh Ngangbam Herojit, and Khelchandra Thongam - Mobile robot navigation using fuzzy logic in static environments. Procedia Computer Science **125** (2018) 11-17. https://doi.org/10.1016/j.procs.2017.12.004
86. Mohamed N., Mohamed B., Lakhmissi C., Vicenc P. - Mobile robot visual navigation based on fuzzy logic and optical flow approaches. International Journal of System Assurance Engineering and Management **10.6** (2019) 1654-1667. https://doi.org/10.1007/s13198-019-00918-2
87. Alexandr S., Thuan P. V., Vaclav K. and Lam P. K. - Optimization of fuzzy logic controller used for a differential drive wheeled Mobile robot. Applied Sciences **11.13** (2021) 6023 1-23. https://doi.org/10.3390/app11136023
88. Claudio Urrea, John Kern and Johanna Alvarado. - Design and Evaluation of a New Fuzzy Control Algorithm Applied to a Manipulator Robot, Applied Sciences **10** (21) (2020) 7482 1-21. https://doi.org/10.3390/app10217482
89. Yong T., Jiaqi Z., Yuanchang L., Tianmiao W., Hegen X., Guotian H. and Dong X.. - Fuzzy PID control method of deburring industrial robots. Journal of Intelligent & Fuzzy Systems **29** (2015) 2447–2455. DOI:10.3233/IFS-151945
90. Jinglei Zhou and Qunli Zhang. - Adaptive Fuzzy Control of Uncertain Robotic Manipulator. Mathematical Problems in Engineering, (2018) 4703492 1-10. https://doi.org/10.1155/2018/4703492
91. Hartono, R., and T. N. Nizar - Speed control of a mobile robot using fuzzy logic controller, IOP Conference Series: Materials Science and Engineering **662** (2) IOP Publishing (2019) 1-7. DOI 10.1088/1757-899X/662/2/022063
92. Saroj Kumar, Dayal R. Parhi - Trajectory planning and control of multiple mobile robot using hybrid MKH-fuzzy logic controller. Robotica **40** (11) (2022) 3952 - 3975. https://doi.org/10.1017/S0263574722000698
93. Krainev A.F., Glazunov V.A. - New mechanisms of relative manipulation. - Problems of mechanical engineering and reliability of machines **5** (1994) 106-117.

94. Khoi P. B. - Principle of compatibility for analysis of forces in mechanisms of relative manipulation. Thesis of PhD. IMASH. RAS. Moscow (1997).
95. Khoi P. B. - Calculation and simulation the program motion of mechanisms of relative manipulation (MRM). J Sci Technol. **3** (2009) 19-28. https://doi.org/10.15625/0866-708X/47/3/1264
96. Khoi Phan Bui - Algorithm and program to calculate the dynamics of parallel robot, Journal of Science and Technology **48** (1) (2010) 33-44. https://doi.org/10.15625/0866-708X/48/1/1084
97. Lung Wen Tsai - Robot Analysis, The Mechanics of Serial and Parallel Manipulators. John Willey & Sons, New York/Toronto, 1999.
98. John J. Craig - Introduction to Robotics, Mechanics and Control. 3rd Edition, Pearson (1989).
99. B. Siciliano, L. Sciavicco, L. Villani, G. Oriolo. - Robotics: Modelling, planning and Control. Springer (2009).
100. Guanrong Chen, Trung Tat Pham. - Introduction to Fuzzy Sets, Fuzzy Logic, and Fuzzy Control Systems. CRC Press, 2001.
101. Michel D. G. - Invariance and Stability of Fuzzy Systems. Journal of mathematical analysis and applications **99** (2) (1984) 299-319. https://doi.org/10.1016/0022-247X(84)90220-8
102. Wong L. K., Frank H. F. L. - Lyapunov-Function-Based Design of Fuzzy Logic Controllers and Its Application on Combining Controllers. IEEE Transactions on Industrial Electronics **45** (3) (1998) 502-509. DOI: 10.1109/41.679009
103. Kandel A., Luo Y., Zhang Q. - Stability analysis of fuzzy control systems. Fuzzy Sets and Systems 105 (1999) 33-48. https://doi.org/10.1016/S0165-0114(97)00234-0
104. James Carvajal, Guanrong Chen, Haluk Ogmen - Fuzzy PID controller: Design, performance evaluation, and stability analysis. Information Sciences **123** (2000) 249-270. https://doi.org/10.1016/S0020-0255(99)00127-9
105. Lam H. K., and Leung F. H. F. - Stability Analysis of Fuzzy Control Systems Subject to Uncertain Grades of Membership. IEEE Transactions on Systems, Man, and Cybernetics Part b: Cybernetics **35** (6) (2005) 1322-1325. DOI: 10.1109/TSMCB.2005.850181
106. Precup R. E., Preitl S. - Stability Analysis of Fuzzy Control Systems. Multivariable Point of View. IFAC. 16th Triennial World Congress, Prague, Czech Republic **38** (1) (2005) 396-401. https://doi.org/10.3182/20050703-6-CZ-1902.01144
107. Radu-E. P., Marius L. T., Stefan P., József K. T., Adrian S. P. - Stability Analysis approach for fuzzy logic control systems with mamdani type fuzzy logic controllers. CEAI **9** (1) (2007) 3-10.
108. Hak K. L. and Frank H. F. L. - Stability Analysis of Fuzzy-Model-Based Control Systems. Springer (2011).
109. Pablo J. P., Nohe R. C. C., Luis T. A., Selene L. C. M. and Jorge A. L. R. - A Lyapunov Analysis for Mamdani Type Fuzzy-Based Sliding Mode Control. IEEE Transactions on Fuzzy Systems **28** (8) (2020) 1-9. DOI: 10.1109/TFUZZ.2019.2923167