www.ijecs.in
International Journal Of Engineering And Computer Science
Volume 12 Issue 08 August 2023, Page No. 25774-25791
ISSN: 2319-7242 DOI: 10.18535/ijecs/v12i08.4744

Control System Design Based Neuro Fuzzy Algorithm for Modified Robotic System PUMA560

Managib N. Abdelgadir[1, 2,] Marghani Fathelrahman[3], Zeinab Mahmoud Yousif[1]

[1] Electrical and Electronics Engineering Department, University of Bahri, Sudan
[2] Mechatronics Engineering Department, Elgraif Sharq Technological College, Sudan Technological University, Sudan
[3] Electrical and Electronics Engineering Department, University of Khartoum, Sudan
* Corresponding author: Managib N. Abdelgadir

Abstract
In robotics, a design of computational system is an important issue which is enabled to represent the functions of robotics and learning the robotic brain to control motions. AI technologies and such neural network enables to to diagnose the error resulting from the movement of the robot by training it to compare the values given to the network through tables in the form of matrix and the real values of the movement of the robot to give the network output to the correct movement of the robot. This paper provides a design of a neural network system for robotics to control the robot movement through a programming and control interface. The evaluation results show the proposed design describing the transit response analysis of the joint actuator (robot motors) to ensure the stability and synchronous will be done.

Keywords: Robotics, Intelligent Motions, Neural Networks, Sensors, Smart Manipulations, PUMA560.

1. Introduction

As a result of the great development of computers and artificial intelligence, robots have been designed. Robots have many advantages in industry which can be represented by increased productivity, lower labor costs, flexibility and ease of programming, completion of work in the shortest period of time, ability to work in hazardous conditions, good investment returns, and freedom of movement in three dimensions in space [1][2]. Also will be find that the most important factor for the development of robotics technology, which helped robots break into the field of industry [3]. It is the discovery of microprocessors, which were able to produce programs capable of implementing coordinated movements of the robot. And when talking about the movement of the robot, it will be mean the workspace, where the workspace refers to the place in which the robot works, as well as the speed of movement [4]. This is determined by several factors such as the weight of the object to be moved and the distance that must be placed in the body during the work cycle [5]. These movements are through the control circuits of robotic systems, and the control systems can work as an open or closed system [6]. The movement in robots is through motors, whether they are dc, servo, or hydraulic motors and etc. The type is selected depending on the required accuracy of the robot.

In order for the robot to be able to work, the artificial robot requires information and data about the surrounding environment, such as the position of the body to be carried, the circumference of these bodies, the angular velocity and moments ... etc. Therefore, sensors are used to transform the form of the information that it receives into a usable output. The robot has both industrial and non-industrial uses [7]. In industrial applications, it is used, for example, in packaging and transportation, spot welding, drilling operations, lifting and lowering operations, fixing and linking, inspection, cutting, printing and assembly works [8]. Also find in production lines in factories, there are fixed platforms built around the production line on which the robot is installed to perform a specific function such as assembly, drilling, Surface finishing and welding. These systems are expensive to install and require a lot of experts and engineering cadres to design and program them. After installing these platforms, it is difficult to modify or reprogram them when the product changes. Therefore, great efforts have been made by engineers and technicians to design unique robots that can move and manipulate [9]. As for non-industrial applications, the importance of robots in practical life comes from the large number and diversity of jobs in life, and from these applications, for example, the field of space technology, hospitals, home services and the environment [10].

In robotic design, The PUMA560 is a fixed base for the arm and most industrial applications require the robot to be mobile to perform a number of tasks across the production line [11]. Moreover, when the robot performs a specific task, this requires the implementation of a certain movement of the robot, and errors may occur in the implementation of this movement depending on modeling, it is difficult to control the series motors in synchronized work to accomplish a certain task, the problem of motion planning for both robot manipulator and mobile robot and the problem controlling of robot manipulator to find the time behavior of the force and torque to be deliver by the joint actuator to ensure the execution if the reference trajectories [12].

A. The Paper Motivations

According to what was discussed above, the objective of this paper is organized as follows:

- A modification of the PUMA 560 robot by adding a mobile robot to connect with the fixed base of the robot arm to be able to perform various tasks.
- The use of two modified PUMA560 robots to synchronize them is due to its importance in industrial fields.
- Diagnostics and treatment of the error of the fixed part of the modified robot PUMA560, which is represented by the movement of the arm at angles, and the movement part, which is represented by the deviation from the path.
- Using the neural network to diagnose the error resulting from the movement of the robot by training it to compare the values given to the network through tables in the form of matrix and the real values of the movement of the robot to give the network output to the correct movement of the robot.

B. Paper Contribution

This paper provides a modified mechanical structure of manipulator robot puma560 from a fixed base (robot manipulator) to a mobile base called mobile manipulator, and these combine give the dexterity of the articulated arm. The other problem of synchronization, motion planning, and controlling the solution of their depending on artificial network based on the controller to diagnose the error by computing the deviation between the reference input and the data provided by the sensor and modified it.

2. Intelligent Algorithms for Robotics Systems

Robotics systems rely on intelligent systems such as fuzzy logic, neural networks and evolutionary strategies to help in the process of designing algorithms for control, process monitoring, and diagnostics for decision-making to reach the best results [13]. Classical logical systems based on hand crafted algorithms usually lack the ability to independently add knowledge to their databases or adapt to a dynamic changing environment, but on the other hand, intelligent systems technologies offer the possibility of independent learning, systematic thinking, adaptability [14][15]. Thus the basic concepts of soft computing techniques are useful for improving complex or undefined problems as well as for the development of intelligent systems and their application to the design, optimization and control of complex and uncertain problems and systems.

2.1. Fuzzy Logic

It is the method by which the degree of affiliation or degree of validity is determined in the event of uncertainty or complexity in many physical systems or processes when the exact analysis of the system is difficult. Knowledge-based systems are defined as systems that have been specifically developed to advance human thinking in solving problems [16]. One of the types of knowledge-based systems is what is known as expert systems and fuzzy logic, which is a set of concepts and techniques that are used as a tool for dealing with qualitative aspects of perception that are in many ways imprecise and ambiguous.

According to the standards of classical mathematics. The science of guess work needs experts to take some decisions in some cases to provide solutions to some problems [17]. This leads us to a concept of smart systems as artificial entities that include a combination of software and hardware that have the ability to acquire and apply knowledge in an intelligent manner and have the capabilities to understand, think, learn, infer and make decisions from Missing information and these smart systems have advantages as the system has the ability to produce depending on some inputs and the nature of the system itself, the inputs to the system may include information in addition to tangible materials and outputs may include decisions in addition to natural products. And this intelligent system has one or more characteristics [18]. These are sensory comprehension, pattern discrimination, appropriation of knowledge, learning and inference from missing information, inference from qualitative and approximate information, ability to deal with unexpected situations, inductive reasoning. In the fuzzy system, knowledge is represented by the if-then rules associated with the fuzzy variables for these rules to be processed by what is called a synthetic inference rule.

2.2.Evolutionary algorithms

Evolutionary algorithms are computational algorithms inspired by the evolutionary process of nature, and evolutionary computing in which the search for the evolution of the solution algorithm is carried out by retaining more appropriate components in the procedure that is analogous to biological evolution. It also mimics the principles of natural selection, which favors stronger species and directs further evolution so that they survive in their environmental conditions [19]. Evolutionary computing has some characteristics including that it can play an important role in the perfect development of the intelligent machine and its self-improvement as it depends on probabilistic reasoning like logic Fuzzy.

2.3. Neural Network

They are artificial neurons connected to each other on a large scale to form a computing network that performs a specific task in the same way as the human brain, through massive processors distributed in parallel [20]. The idea came from the bio-architecture of neurons in the human brain. One of the main characteristics of neural networks is their ability to approximate nonlinear functions, since intelligent machines include a special class of nonlinear decision-making [21]. Any neural network that includes a number of nodes organized in layers that are connected through weight elements are called neural connections.

When will be want to build a neural network to perform some task, will be must first determine the number of nodes to be used, the appropriate types of nodes, and how to connect the nodes to form the network. Then will be initialize the network weights and train the weights using a learning algorithm applied to a set of training examples for the task [22]. And through the use of examples, will be must know how to encode the examples in terms of network inputs and outputs.

Usually the network is flagged by updating the weights and the weights are modified to try to make the network's input and output behavior consistent with the behavior of the environment that provides the input [23]. In practice, neural network implementations are in software and use simultaneous control to update all modules in a fixed sequence. Finally, mathematically, neural networks contain a variety of mathematical symbols that are used as algorithms, which are summarized in Table 1.

Table 1. Neural networks contain a variety of mathematical symbols

Notation	Meaning
$\boldsymbol{a}_i$ a_i	The output of the unit The inputs to unit *i*
g g'	Activation function Derivative of the activation function
E rri Err^e	Error unit i Error for example *e*
I_i I r	a unit i in the input layer Vector of activations of all input units Vector of inputs for example *e*
in_i	Weighted sum of inputs to unit i
N	Total number of units in the network
O O_i O	The single output unit of a perceptron a unit i in the output layer Vector of all units in the output layer
t	Threshold for a step function
T T ***T***	Target output for a perceptron Target vector when there are several output units Target vector for example *e*
$W_{j.i}$ W_i W, W	Weight on the link from unity to unit *i* Weight from unit r to the output in a perceptron Vector of weights leading into unit *i* Vector of all weights in the network

3. The Methodologies

The proposed study provides a design and implementation of a software to control the motion of a modified robotic puma560 in applications. And the hardware includes modified robotic puma560, the essential components of robot are a mechanical system with locomotion apparatus (wheels) and manipulation apparatus (mechanical arm), a sensor to sense the motion of the robot and transfer [24]. This data as signals to the controller to be manipulated, PC MATLAB used as software contain the intelligent neural network(ANN) that receives the data from the sensor connected to the robot to send to the controller as the input to the neural network to process and send the output of the network to the control unit by interface (VSPE) which connected between PC and the controller to move the motors of robot , controller unit contain programming to control the modified robots PUMA560 motion [25]. After that, show the results as a curve in MATLAB which describes the transit response analysis of the joint actuator (motors of robot) to ensure the stability and synchronous will be done [26].

3.1. Neuro Fuzzy Algorithm

In building a robotic system by using Neuro fuzzy based controller to diagnosis with modified, Robotic puma 560 it has to look into its main components. The project consists of four main components, the artificial neural network (ANN) unit, control unit (Arduino mega), sensor, and robots, as shown in Figure 1.

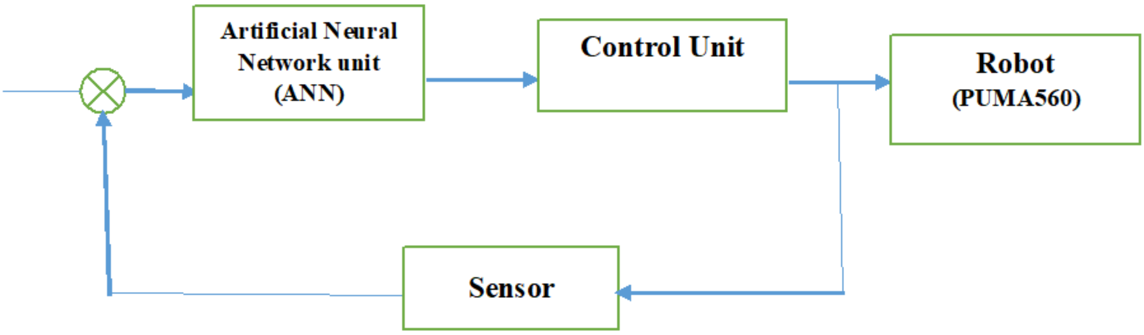

Fig.1. the block diagram of components.

The artificial neural network (ANN) unit in this project will be used data from sensor throw Adriano to treatment and send the output to the control (Adriano) to move the robot [27]. First, the neural network is created by choosing the number of input and output nodes and the hidden layer. Second, the network is trained with certain data [28]. The controller based neural network is then used to diagnose the error by calculating the deviation between the reference input and the data provided by the sensor and adjusting it according to the network training previously.

3.2. Control Unit Design

The control unit will be used in this project to transmit and receive data (from ANN and sensors) by the programmed found it. The control units used in the project design consists mainly Arduino mega, motor driver and LCD. Controller (Arduino mega) is consists of an integrated central processing unit, with which the computing unit and the sequencing unit, and the program memory used for the storage program [29]. Additional element data memory such as RAM for data storage, ROM for program storage, flash memory for permanent data storage, peripherals, and I/O interfaces. Basically, all controllers are built on the same concept as shown in the figure2.

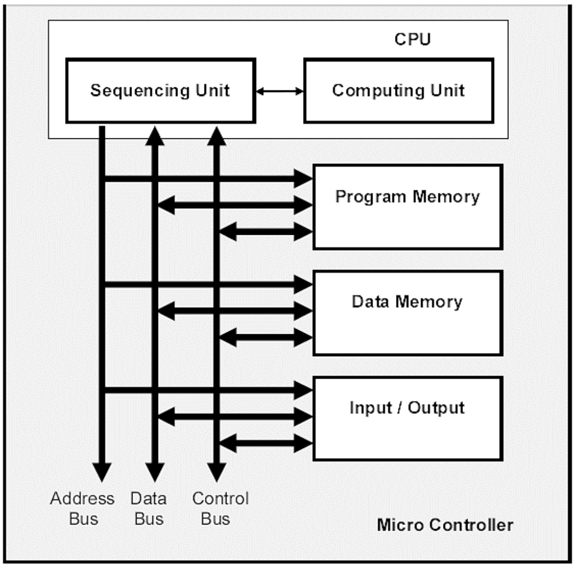

Fig.2. Basic structure of a controller

Arduino Mega 2560 will be used in this project and it is works at 16MHz. The microcontroller has 54 digital I/O ports, 16 analog inputs, 4 UARTs (hardware serial ports), a USB connection, a power jack, an ICSP header, and a reset button. The block diagram of The MPU-6050 that used in design is comprised of the following key blocks and functions as shown in Figure 3.

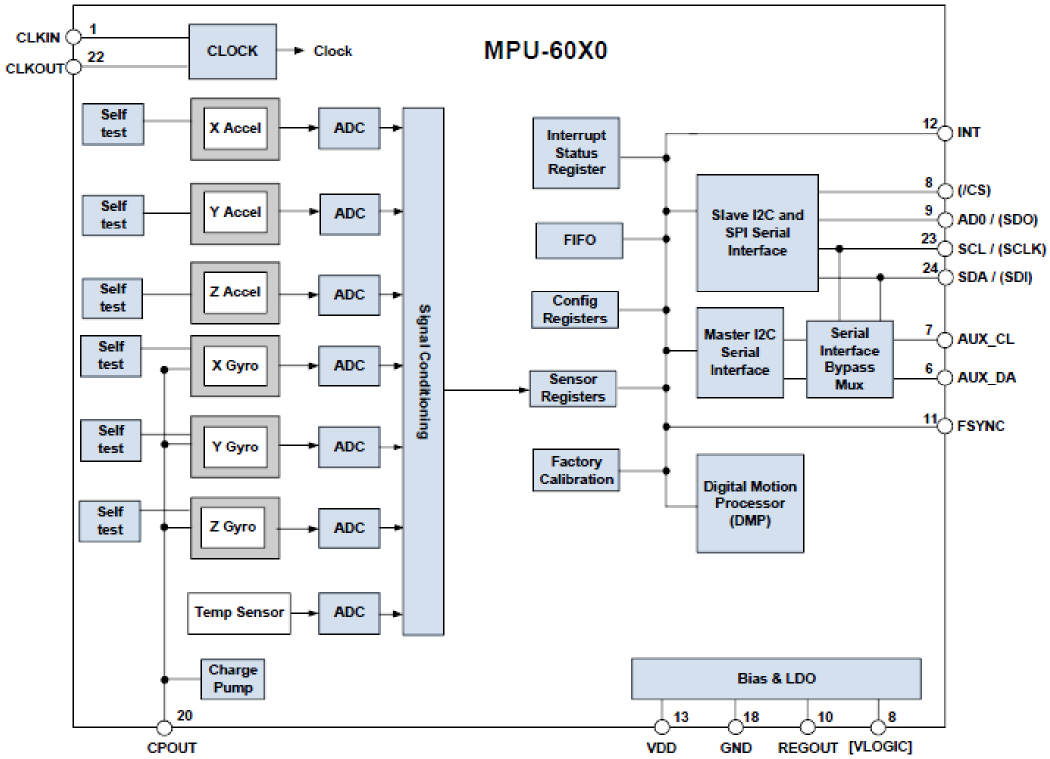

Fig.3. MPU6050 Block Diagram

3.3. The Robotics Main Devices

The robot will be used in the proposed study is modified puma560 It's representing the mechanical component of the design. Robots are distinguished from each other by the effectors and sensors with which they are equipped. Two Robotics will be used for Synchronous System Analysis in Intelligent Industrial Field and adaptation of a system-modified robotic puma 560 6dof to increase the efficiency of the system's performance in industrial field. And modified in robotics by adding Mobile robot (wheels) to move in a specific path from point to point at synchronized [30]. After that, the arm of the robotics will be operation depending on the smart algorithms in practice after training them in an intelligent computer environment. It is consisting of a screwdriver or other tool, a welding gun, paint sprayer, or a gripper. The motor of robots will be used for the motion of the mobile robot is A DC motor and servo motor to motion arm of robot [31].

It Will be studying these motors and their equations because through them will can know the analysis of system stability and synchronization by studying the time behavior that shows the safety of movement of the robots that are related to the torque and strength of the motors. The modeling of these motors explains the bellow.

A. DC Motor Modeling of mobile robot

It is an electrical machine that converts electrical energy (DC) into mechanical energy (motion). It is composed of the stator with the external structure and magnetic poles, the rotor, the union member, the carbon brushes, the two side covers and the connecting plate [32]. DC motors are one of the actuators in the control systems and it provides us with rotational movement, so it can be connected to wheels, pulleys, gears and the equivalent electrical circuit as shown in the figure4.

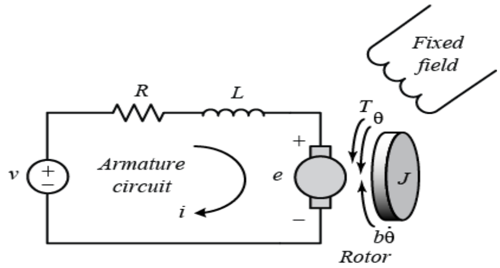

Fig.4. Model of DC motor circuit

In this design, the system input will be assumed to be the source voltage (V) applied to the motor, while the output is shaft rotational speed d (theta)/dt. It is assumed that the friction torque is proportional to the angular velocity of the shaft. In general, the torque generated by a DC motor is proportional to the armature current and the strength of the magnetic field. In this design will be assume that the magnetic field is constant and, therefore, the armature torque is proportional to the armature current i by a constant factor K t as shown in the equation below. This is referred to as a motor controlled motor.

$$T = k_t i \tag{1}$$

The back emf, e, is proportional to the angular velocity of the shaft by a constant factor K_e

$$e = k_e \theta' \tag{2}$$

In SI units, the motor torque and back emf constants are equal, that is, $K_t = K_e$; therefore, there will be used K to represent both the motor torque constant and the back emf constant.

From the figure above, it will be can derive the following governing equations based on Newton's 2nd law and Kirchhoff's voltage law.

$$J\ddot{\theta} + b\dot{\theta} = k\, i \tag{3}$$

$$L\frac{di}{dt} + R\, i = V - k\,\dot{\theta} \quad . \tag{4}$$

Applying the Laplace transform, the above modeling equations can be expressed in terms of the Laplace variable s.

$$s(Js + b)\theta(s) = K\, I(s) \tag{5}$$

$$(Ls + R)I(s) = K\, I(s) \tag{6}$$

We arrive at the following open-loop transfer function by eliminating I(s) between the two above equations, where the rotational speed is considered the output and the armature voltage is considered the input. [14]

$$P(s) = \frac{\dot{\theta}(s)}{V(s)} = \frac{K}{(Js+b)(Ls+R)+K^2} \tag{7}$$

B. Servo Motor Modeling arm robot

Servo motor is used for position or speed control in closed-loop control systems. It has implemented a proportional integral neuro fuzzy inference system, respectively, for the variable working situations to the simulation model which has been prepared at the MATLAB programmers for improvement servomotor performance [33]. The equivalent circuit diagram of the servo motor is presented in Figure 5, The armature is modeled as a circuit with resistance, R_a connected in series inductance, L_a and a voltage source, $V_b(t)$ representing the back emf of the armature when the rotor rotates.

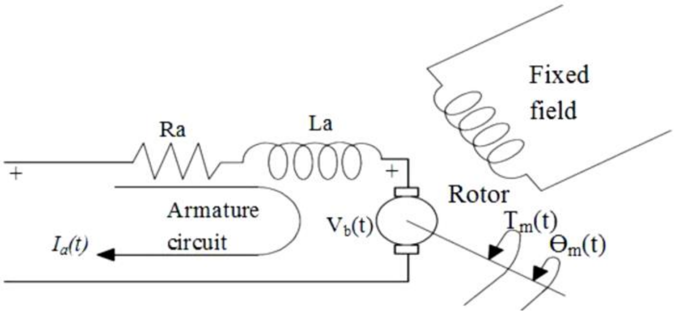

Fig.5.Model of servomotor circuit

C. Sensor

The sensors will be used in this project to sense the input of motion to transfer data to signal to the controller to manipulate. There are two types of sensor that will be used, IR sensor for the wheel car and MPU sensor for the angle position of the arm of robotics [33].

3.4. Software and Design Model

The software will be used for simulation is MATLAB, Arduino code, Proteus, and VSPE. MATLAB program is used to insert the data code, and the train of ANN. Arduino code is used for writing the language code and then transform the code to a hex file to download the program on the chip by Arduino at mega 2560. Proteus program software to select the components from liberally and connected them to simulation [34]. The VSPE program is used to interface between the MATLAB program and the controller.

The robotic circuit designed by Proteus program to represent the model of the project to control the two robotics simulation. It consists of 8 sensors, 5 driver motors, a controller (Arduino mega), 14 actuators (motor of robots), VSPE interface and LCD as shown in Figure 6. The circuit's connection microcontroller connected to sensor, drivers and VSPE interface and all devices connected to V_{cc} and ground.

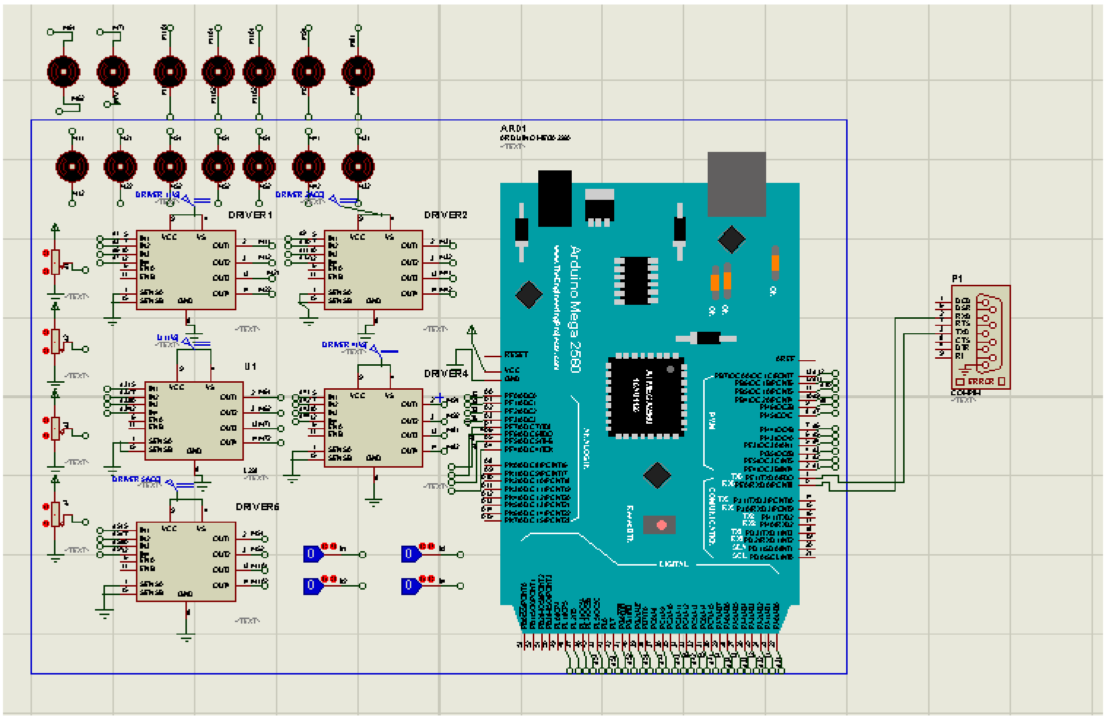

Fig.6. The circuit diagram of the model

The movement of robot A and robot B is done simultaneously by entering the inputs by sensors as well as outputs represented in speed and angle, and these inputs and outputs are represented by a matrix and they are called the input matrix and the output matrix. These arrays are entered by using the MATLAB program and it is named (Data Code). These matrices will be entered into the intelligent network (ANN), which is a network that is also used through the use of a special code (Train code) in MATLAB that enters the data and trains on it through a specific algorithm to produce the required output to be connected to the controller to run the robot in the required path and at the required angles. So the working theory can be summarized through the following steps: the input and output matrix of the robot paths, training matrices by intelligent network (ANN), using the controller for the input and output variables and the simulation of the model and result.

The input and output parameters represents the sensors as input and motors as output. After entering the input and output matrices in the MATLAB program, these inputs are processed through the neural network and trained through a specific program written in MATLAB and called the code train by entering the input values that were read from the sensors and thus entered into the neural network [35]. The sensors may give several readings when the robot is moving, when an error may occur in the movement or angles of robot. Only by training the network, it chooses the correct value or approximately to the reading sensor to come out as a perfect output from the network to the controller, which in turn gives the command to the driver connected with it to move the robot. And the steps of training in MATLAB as shown in the following Figure 7 and 8.

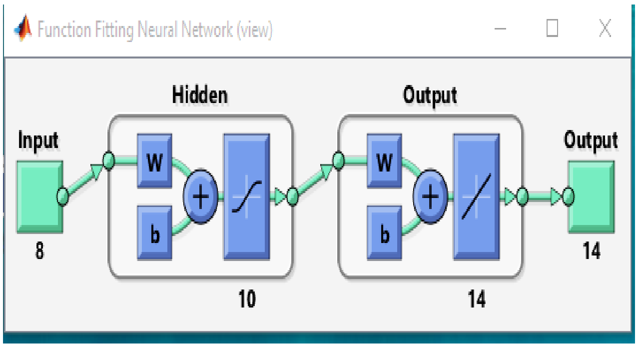

Fig.7. Training matrices by intelligent network(ANN), explain the train when inserting the data code and train code after run in MATLB

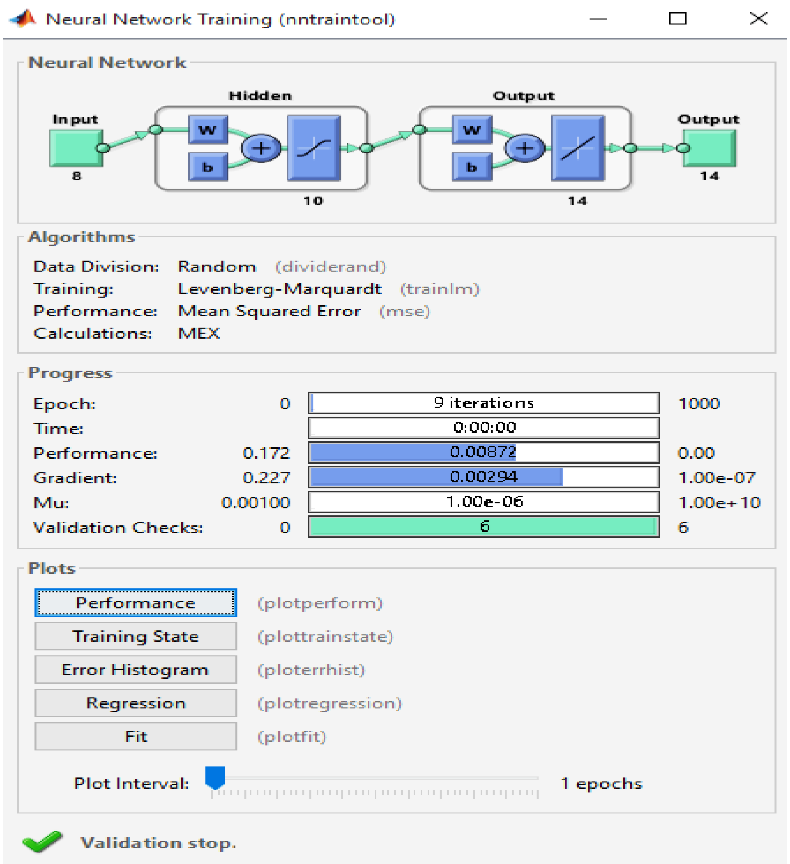

Fig.8. explain the performance of ANN and error

After the intelligent neural network output, the ATmega2560 controller receives the network output in the form of data for the synchronous movement of the robot and the robot arm through the inputs of the controller. Through the Arduino program used in the microcontroller, a program called Arduino code was prepared in which commands are given to the inputs and outputs and the relationship between them is created, thus controlling the robot's motors and its movement.

3.5. Simulation Implementation

A simulation system was made through the use of MATLAB and Proteus programs, and the link between the two programs was done through the VSPE program, the system of the robotic is divided into five modules, and these are PC for (neural network in MATLAB), controller module, driver Sensor, VSPE interface and LCD operation of the simulation as shown in flowcharts in figure 9.

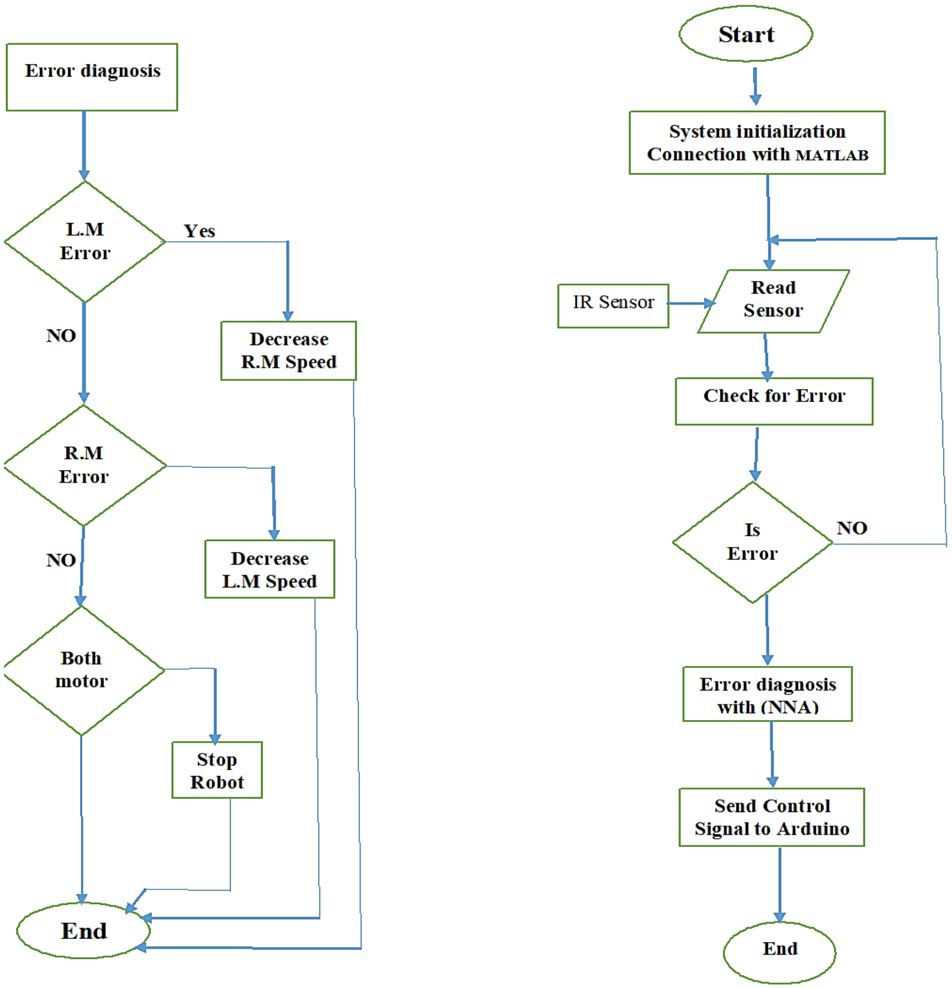

Fig.9. Flowcharts of the operation system

4. Results and Discussion

The result after simulating the designed circuit diagram of the model, will be noted when insert the input sensor, by taking the scenario of performing the motors for mobile robot. According to the given income to produce an output that gives this synchronous movement that produces these curves to show the response to these motors as shown in figures 10 and 11.

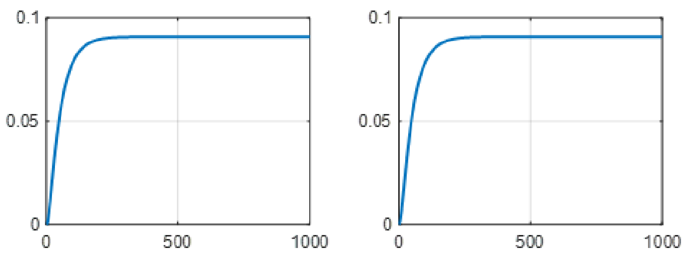

Fig.10. The performance of the robot A with two motors (M1&M2)

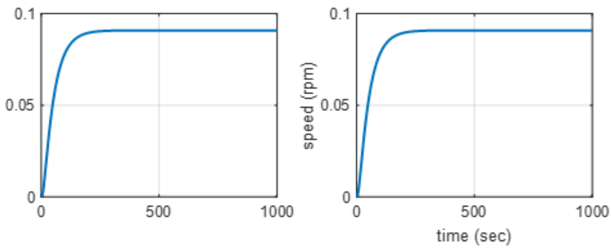

Fig.11. the performance of robot B with two motors (M8&M9)

These results which describe the transit response analysis of robot motors to ensure the stability and synchronous will be done in two mobile Robots (A and B). These results will be presented in the above curves in the tables to help us in the analysis as shown in table 2.

Table 2. The performance behavior of both mobile robots

Performance	**Rise time**	**Setting Time**	**Setting Min**	**Setting Max**	**overshoot**	**Undershoot**	**peak**	**Peak Time**
Robot A M1	1.0161	1.8471	0.0818	0.0908	0	0	0.0908	10
Robot A M2	1.0161	1.8471	0.0818	0.0908	0	0	0.0908	10
Robot B M8	1.0161	1.8471	0.0818	0.0908	0	0	0.0908	10
Robot B M9	1.0161	1.8471	0.0818	0.0908	0	0	0.0908	10

The table shows the explanation of the response of the motors that appeared when the two robots were moving together, as shown in figure 10, that there was synchronization in the movement and therefore no change occurred in the parameters. Robot A has two motors for movement (M1, M2) and robot B also has two motors (M8, M9) that work together.

In the case of the robot arm, after the two robots stop together at a certain meeting point, then the movement of the arm is done according to the given input. The output is produced through the movement of the angles of the arm and thus the response of the arm motors is as shown in figures 12, 13, 14, 15, and 16 for each motor alone.

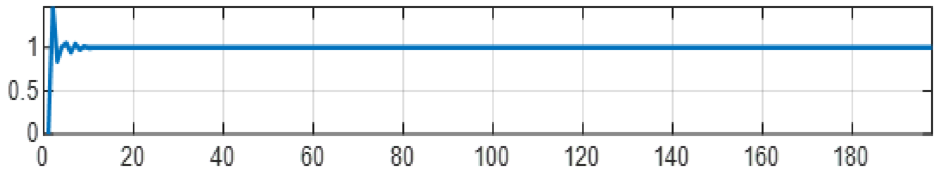

Fig.12. The performance of Arms robot A (motor M3)

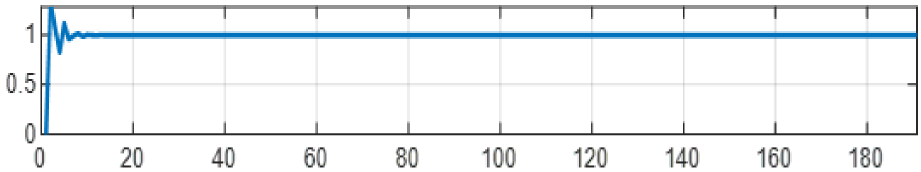

Fig.13. The performance of Arms robot A (motor M4)

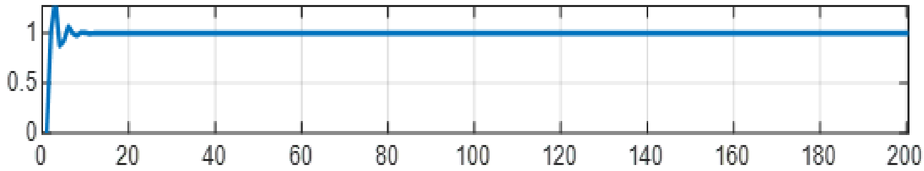

Fig.14. The performance of Arms robot A, motor M5)

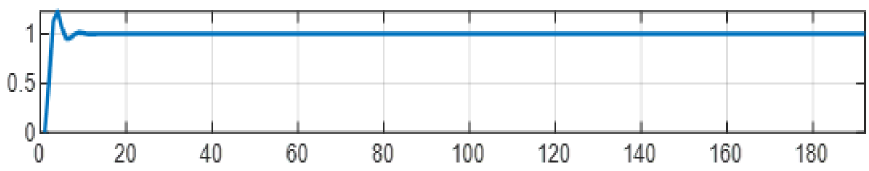

Fig.15. The performance of Arms robot A (motor M6)

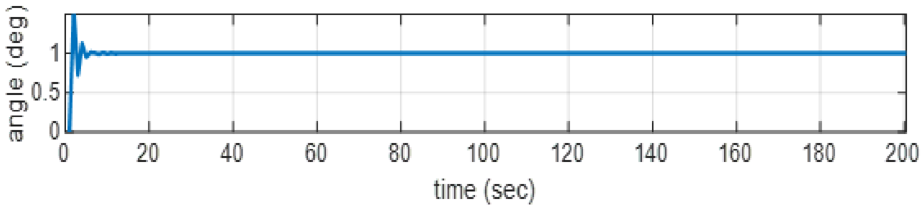

Fig.16. The performance of Arms robot A (motor M7)

To explain the output that resulted from the above figure, the results which describe the transit response analysis to the joint actuators (M3,M4,M5,M6and M7) of Arm Robot A to ensure the stability and synchronous will be done in Arm Robot A .The same understanding is true for the Robot B To explain the output that resulted from the figures 17, 18, 19, 20, and 21 below the results which describe the transit response, analysis to joint actuator (M10,M11,M12,M13and M14) of Arm Robot B to ensure the stability and synchronous will be done in Arm Robot B.

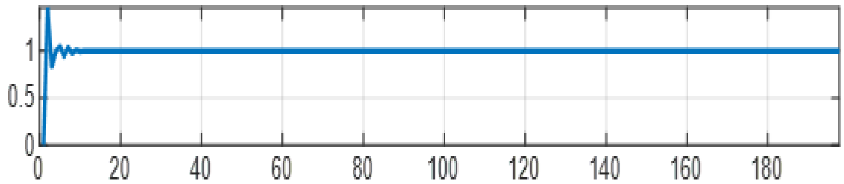

Fig.17. The performance of Arm robot B (motor M10)

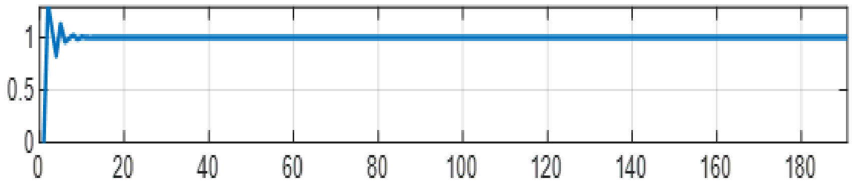

Fig. 18, The performance of Arms robot B (motors M11)

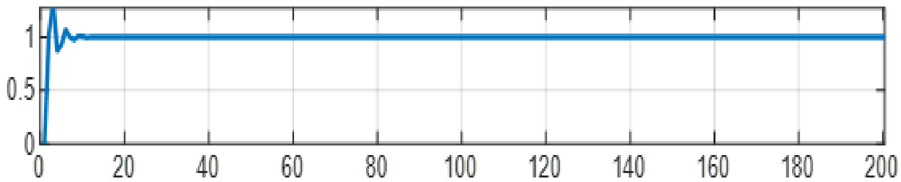

Fig.19. The performance of Arm robot B (motor M12)

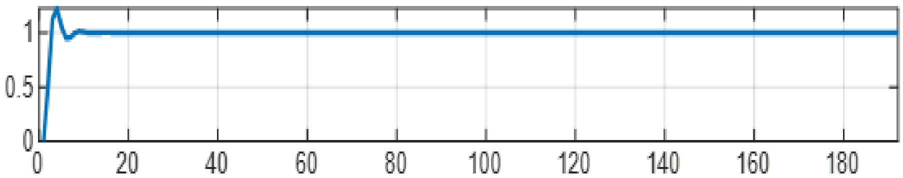

Fig.20. The performance of Arm robot B (motor M13)

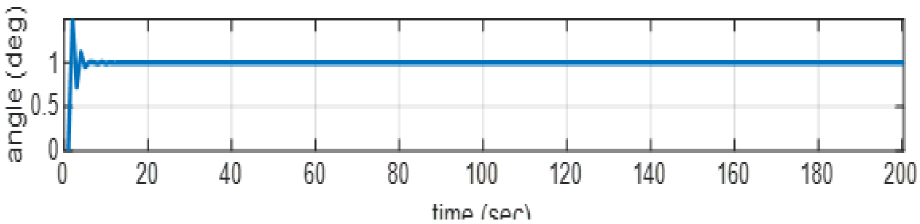

Fig.21. The performance of Arm robot B (motor M14)

These results will be presented in the form of curves in the form of values in the tables to help us in the analysis as shown in table 3.

Table 3. The performance behavior of both arm robots

Performance	Rise time	Setting Time	Setting Min	Setting Max	overshoot	Undershoot	peak	Peak Time
Robot A M3	0.0053	0.0707	0.7808	1.5133	51.3252	0	1.5133	0.0100
Robot A M4	0.0063	0.0716	0.7954	1.2642	26.4194	0	1.2642	0.0100
Robot A M5	0.0073	0.0647	0.8224	1.303	30.2273	0	1.3023	0.0200
Robot A M6	0.0143	0.0659	0.9497	1.2247	22.4748	0	1.2247	0.0300
Robot A M7	0.0052	0.0444	0.7328	1.5405	54.0505	0	1.5405	0.0100
Robot B M10	0.0054	0.0713	0.8389	1.4798	47.9805	0	1.4798	0.0100
Robot B M11	0.0060	0.0702	0.8253	1.3282	32.8160	0	1.3282	0.0100
Robot B M12	0.0079	0.0714	0.8725	1.3412	34.1223	0	1.3412	0.0200
Robot B M13	0.0143	0.0659	0.9497	1.2247	22.4748	0	1.2247	0.0300
Robot B M14	0.0052	0.0447	0.7258	1.5445	54.4474	0	1.5445	0.0100

For the first scenario, the mobile robot (wheel), when entering the input values of the IR sensor according to the matrix values in the system, the working theory is described in Table 4.

Table 4. Working model of mobile robot (wheel)

IR Sensor(S_1,S_2,S_3,S_4)		Working model
$S_{3,4}$	$S_{1,2}$	
0	0	When Sensor(IR_1,IR_2,IR_3,IR_4) take this value the motor of two robot A and Robot B is operation at the same time(synchronous)
1	0	When Sensor(IR_1,IR_2,IR_3,IR_4) take this value the motor of robot A (M_{61},M_{62}) as one side only operated and Robot B (M_{11},M_{12}) is one side only operated this case happens when the robot deviation in right direction.
0	1	When Sensor(IR_1,IR_2,IR_3,IR_4) take this value the motor of robot A (M_{71},M_{72}) as one side only operated and Robot B (M_{21},M_{22}) is one side only operated this case happens when the robot deviation in Left direction.
1	1	When Sensor (IR1, IR2, IR3, IR4) take this value the motor of two robot A and Robot B is operation at the same time stopped this case happens means that the robot has arrived to destination.

As well, depending on the Table 4 explained above, the two robots are moved together when the neural network receives the reading of the sensors and gives its output to the controller to move the motors together, as well as, it is producing curves that show the performance of these motors.

When comparing the difference between the performance of the robots A a and B through the curve, values resulting from the movement (by subtracting the values of the parameters from each other, that is, for each motor of the robot with the motor of the robot that corresponds to the other direction), will be found that the values of the movement are fixed and therefore all take zero, as shown in table 5 and this indicates the stability of the system and that there is no Error and synchronization check will be find that the error rate between the two robots together is completely absent, and thus the principle of stability and synchronization is achieved.

Table 5. Comparing Performance between two mobile robots

Comparing Performance	Rise time	Setting Time	Setting Min	Setting Max	overshoot	Undershoot	peak	Peak Time
Robot(A-B) (M1/ M8)	0	0	0	0	0	0	0	0
Robot(A-B) (M2/ M9)	0	0	0	0	0	0	0	0

For arm robot. All the angle when the mobile robot moves as synchronizations, it becomes constant .it is moved when the mobile robot is stopped. These moving depending on the angle insert from MPU sensor and appear in the command window in MATLAB. For determination of the results which describe the transit response, analysis of the joint actuator (robot motors) to ensure the stability and synchronous will be done then comparing the results between the two. The comparison is difference between the performance of the robots A and B as shown in Table 6 through the curve values resulting from the movement (by subtracting the values of the parameters from each other, that is, for each motor of the robot with the motor of the robot that corresponds in the other direction).

Table 6. Comparing Performance between two-Arm robots

Comparing Performance	Rise time	Setting Time	Setting Min	Setting Max	overshoot	Unders hoot	peak	Peak Time
Robot(A-B) (M3/M10)	0.0001	0.0006	0.0581	0.0335	3.3447	0	0.0335	0
Robot(A-B) (M4/M11)	0.0003	0.0014	0.0299	0.064	6.3966	0	0.064	0
Robot(A-B) (M5/M12)	0.0006	0.0067	0.0501	0.0389	3.895	0	0.0389	0
Robot(A-B) (M6/M13)	0	0	0	0	0	0	0	0
Robot(A-B) (M7/M14)	0	0.0003	0.007	0.004	0.3969	0	0.004	0

Will be find that the error rate between the two robots in table 6 is very small, thus achieving the principle of stability and synchronization

5. Conclusion

The conclusion of this research is that the puma560 robot has been modified by adding a mobile robot to connect with the fixed base of the robot arm to be able to perform various tasks in industrial applications and the use of two modified PUMA560 robots to synchronize them due to its importance in industrial fields. In addition, two modified robots were used to study the synchronization between them in the movement of the moving part of the robot, represented by the vehicle, where the two robots move with each other at the same time, as well as in the motion of the fixed part represented by the arm, where the two arms move at specific angles also at the same time after the two robot stopped moving at a specified point. These movements may cause errors, so A smart neural network has been designed and trained to diagnose and modify errors to give it to the controller to work on the movement of the robot as required. Finally, the determination of the results which describe the transit response, analysis of the joint actuator (robot motors) to ensure the stability and synchronous will be done then comparing the results between the two robots.

References

1. F Rubio, Valero F, Llopis-Albert C. A review of mobile robots: Concepts, methods, theoretical framework, and applications. International Journal of Advanced Robotic Systems. 2019;16(2). doi:10.1177/1729881419839596
2. Sima Violeta , Ileana Georgiana Gheorghe , Jonel Subi´c and Dumitru Nancu, Influences of the Industry 4.0 Revolution on the Human Capital Development and Consumer Behavior: A Systematic Review, Sustainability 2020, 12, 4035; doi:10.3390/su12104035

3. Ahmadzadeh Hossein, Ellips Masehian, Modular robotic systems: Methods and algorithms for abstraction, planning, control, and synchronization, Artificial Intelligence, Volume 223, 2015, Pages 27-64, ISSN 0004-3702, https://doi.org/10.1016/j.artint.2015.02.004.
4. Ljasenko Spartak, Niels Lohse & Laura Justham , Dynamic vs dedicated automation systems - a study in large structure assembly, Production & Manufacturing Research, 8:1, 35-58, DOI: 10.1080/21693277.2020.1737591
5. FJG Silva, Soares MR, Ferreira LP, Alves AC, Brito M, Campilho RDSG, Sousa VFC. A Novel Automated System for the Handling of Car Seat Wires on Plastic Over-Injection Molding Machines. Machines. 2021; 9(8):141. https://doi.org/10.3390/machines9080141
6. N Nordin, Xie SQ, Wünsche B. Assessment of movement quality in robot- assisted upper limb rehabilitation after stroke: a review. J Neuroeng Rehabil. 2014 Sep 12;11:137. doi: 10.1186/1743-0003-11-137. PMID: 25217124; PMCID: PMC4180322.
7. A Akbari, Haghverd F and Behbahani S, Robotic Home-Based Rehabilitation Systems Design: From a Literature Review to a Conceptual Framework for Community-Based Remote Therapy During COVID-19 Pandemic. *Front. Robot. AI* 8:612331. doi: 10.3389/frobt.2021.612331
8. S Hernandez-Mendez, Palacios-Hernandez ER, et al Design and Implementation of Composed Position/Force Controllers for Object Manipulation. *Applied Sciences*. 2021; 11(21):9827. https://doi.org/10.3390/app11219827
9. H Engemann, Cönen P, Dawar H, Du S, Kallweit S. A Robot-Assisted Large-Scale Inspection of Wind Turbine Blades in Manufacturing Using an Autonomous Mobile Manipulator. *Applied Sciences*. 2021; 11(19):9271. https://doi.org/10.3390/app11199271
10. A Zhang, Koyama K, Wan W, Harada K. Manipulation Planning for Large Objects through Pivoting, Tumbling, and Regrasping. *Applied Sciences*. 2021; 11(19):9103. https://doi.org/10.3390/app11199103
11. Arshad A., S. Badshah and P. K. Soori, "Design and fabrication of smart robots," *2016 5th International Conference on Electronic Devices, Systems and Applications (ICEDSA)*, 2016, pp. 1-4, doi: 10.1109/ICEDSA.2016.7818518.
12. LM Muratori, Lamberg EM, Quinn L, Duff SV. Applying principles of motor learning and control to upper extremity rehabilitation. J Hand Ther. 2013 Apr-Jun;26(2):94-102; quiz 103. doi: 10.1016/j.jht.2012.12.007. PMID: 23598082; PMCID: PMC3773509.
13. J.M Box-Steffensmeier,., Burgess, J., Corbetta, M. et al. The future of human behaviour research. Nat Hum Behav 6, 15–24 (2022). https://doi.org/10.1038/s41562-021-01275-6
14. Chen Ching-Han, Chien-Chun Wang, et al, "Fuzzy Logic Controller Design for Intelligent Robots", Mathematical Problems in Engineering, vol. 2017, Article ID 8984713, 12 pages, 2017. https://doi.org/10.1155/2017/8984713
15. Raguraman S. M., D. Tamilselvi and N. Shivakumar, "Mobile robot navigation using Fuzzy logic controller," *2009 International Conference on Control, Automation, Communication and Energy Conservation*, 2009, pp. 1-5.
16. A Saffiotti,., Ruspini, E.H., Konolige, K, Using Fuzzy Logic for Mobile Robot Control. In: Zimmermann, HJ. (eds) Practical Applications of Fuzzy Technologies. The Handbooks of Fuzzy Sets Series, vol 6. Springer, Boston, MA. 1999, https://doi.org/10.1007/978-1-4615-4601-6_5
17. Cembrano G, C Torras, G Wells, Neural networks for robot control, Annual Review in Automatic Programming, Volume 19, 1994, Pages 159-166, ISSN 0066-4138, https://doi.org/10.1016/0066-4138(94)90059-0.
18. Thomas jith and John Hedley, FumeBot: A Deep Convolutional Neural Network Controlled Robot, Robotics 2019, 8, 62; doi:10.3390/robotics8030062

19. Bozek P, Karavaev YL, Ardentov AA, Yefremov KS. Neural network control of a wheeled mobile robot based on optimal trajectories. International Journal of Advanced Robotic Systems. 2020;17(2). doi:10.1177/1729881420916077
20. Nguyen H. -T. and C. C. Cheah, "Analytic Deep Neural Network-Based Robot Control," in *IEEE/ASME Transactions on Mechatronics*, vol. 27, no. 4, pp. 2176-2184, Aug. 2022, doi: 10.1109/TMECH.2022.3175903.
21. S.G Tzafestas,. (1995). Neural Networks in Robot Control. In: Tzafestas, S.G., Verbruggen, H.B. (eds) Artificial Intelligence in Industrial Decision Making, Control and Automation. Microprocessor-Based and Intelligent Systems Engineering, vol 14. Springer, Dordrecht. https://doi.org/10.1007/978-94-011-0305-3_11
22. Jiang Yiming, Chenguang Yang, Jing Na, et al, "A Brief Review of Neural Networks Based Learning and Control and Their Applications for Robots", Complexity, vol. 2017, Article ID 1895897, 14 pages, 2017. https://doi.org/10.1155/2017/1895897
23. H Deng, Xia Z, Weng S, Gan Y, Fang P, Xiong J. A motion sensing-based framework for robotic manipulation. Robotics Biomim. 2016;3(1):23. doi: 10.1186/s40638-016-0056-9. Epub 2016 Dec 9. PMID: 28018838; PMCID: PMC5148786.
24. K Panayiotou, Tsardoulias E, Zolotas C, et al. A Framework for Rapid Robotic Application Development for Citizen Developers. Software. 2022; 1(1):53-79. https://doi.org/10.3390/software1010004
25. Masoud Mohammad, Yousef Jaradat, Ahmad Manasrah, Ismael Jannoud, "Sensors of Smart Devices in the Internet of Everything (IoE) Era: Big Opportunities and Massive Doubts", Journal of Sensors, vol. 2019, Article ID 6514520, 26 pages, 2019. https://doi.org/10.1155/2019/6514520
26. C Dai Y, Xiang, Qu W, Zhang Q. A Review of End-Effector Research Based on Compliance Control. *Machines*. 2022; 10(2):100. https://doi.org/10.3390/machines10020100
27. Dilip Golda, Ramakrishna Guttula, Sivaram Rajeyyagari, et al, "Artificial Intelligence-Based Smart Comrade Robot for Elders Healthcare with Strait Rescue System", Journal of Healthcare Engineering, vol. 2022, Article ID 9904870, 12 pages, 2022. https://doi.org/10.1155/2022/9904870
28. S-M Kim, Choi Y, Suh J. Applications of the Open-Source Hardware Arduino Platform in the Mining Industry: A Review. *Applied Sciences*. 2020; 10(14):5018. https://doi.org/10.3390/app10145018
29. Hrishikesh Mr. R. Shirke, Dr. Prof. Mrs. N. R. Kulkarni, Mathematical Modeling, Simulation and Control of Ball and Beam System, INTERNATIONAL JOURNAL OF ENGINEERING RESEARCH & TECHNOLOGY (IJERT) Volume 04, Issue 03 (March 2015), http://dx.doi.org/10.17577/IJERTV4IS030879
30. M Xiloyannis, Annese E, Canesi M, et al,. Design and Validation of a Modular One-To-Many Actuator for a Soft Wearable Exosuit. Front Neurorobot. 2019 Jun 18;13:39. doi: 10.3389/fnbot.2019.00039. PMID: 31275129; PMCID: PMC6591529.
31. S N. Paine,. Oh and L. Sentis, "Design and Control Considerations for High-Performance Series Elastic Actuators," in IEEE/ASME Transactions on Mechatronics, vol. 19, no. 3, pp. 1080-1091, June 2014, doi: 10.1109/TMECH.2013.2270435.
32. JU Liceaga-Castro, Siller-Alcalá II, González-San Román JD, Alcántara-Ramírez RA. PI Speed Control with Reverse Motion of a Series DC Motor Based on the Noise Reduction Disturbance Observer. *Actuators*. 2022; 11(5):117. https://doi.org/10.3390/act11050117
33. Baoming G., L. Jihong and A. T. de Almeida, "Fuzzy neural network control for robot manipulator directly driven by switched reluctance motor," *9th IEEE International Conference on Cognitive Informatics (ICCI'10)*, 2010, pp. 573-577, doi: 10.1109/COGINF.2010.5599676.

34. Baoming Ge, and Aníbal T. de Almeida. "Fuzzy Neural Network Control for Robot Manipulator Directly Driven by Switched Reluctance Motor." IJCINI vol.5, no.3 2011: pp.86-98. http://doi.org/10.4018/ijcini.2011070106