UNIVERSITY OF MISKOLC
FACULTY OF MECHANICAL ENGINEERING AND INFORMATICS

ARTIFICIAL INTELLIGENCE AND OPTIMIZATION ALGORITHMS FOR DESIGN AND IMPLEMENTATION OF A ROBOTIC PLATFORM

Booklet of PhD Theses

PREPARED BY:
HAZIM NASIR GHAFIL
Engineering of mechanics … (BSc),
Engineering of applied mechanics … (MSc)

ISTVÁN SÁLYI DOCTORAL SCHOOL
TOPIC FIELD OF DESIGN OF MACHINES AND STRUCTURES
TOPIC GROUP OF DESIGN OF MECHATRONICS SYSTEMS

HEAD OF DOCTORAL SCHOOL
PROF. DR. GABRIELLA BOGNÁR VADÁSZNÉ
DSC, FULL PROFESSOR

HEAD OF TOPIC GROUP
PROF. DR. GABRIELLA BOGNÁR VADÁSZNÉ
DSC, FULL PROFESSOR

SCIENTIFIC SUPERVISOR
PROF. DR. KÁROLY JÁRMAI
DSC, FULL PROFESSOR

Miskolc
2022

JUDGING COMMITTEE

chair: **Prof. Dr. Gabriella Bognár Vadászné,**
DSc, Full Professor (University of Miskolc)

secretary: **Dr. habil. György Kovács**
PhD associate professor (University of Miskolc)

members: **Dr. Károly Széll**
PhD associate professor (University of Óbuda)

Dr. György Hegedűs
PhD associate professor (University of Miskolc)

Dr. Péter Tamás Szemes
PhD associate professor (University of Debrecen)

OFFICIAL REVIEWERS

Prof. Dr. János Száva
PhD professor (Transilvania University of Brasov)

Dr. Tamás Szabó
PhD associate professor (University of Miskolc)

1. INTRODUCTION

1.1 PRELUDE

Nowadays, robots are playing a significant role in all aspects of human life [1-3] because of the human tendency to fulfil his needs with low cost, high quality, and fast production rates, which may be difficult just by workers as well as these robots is preferred especially in some difficult working environments, which should undergo for a severe risk assessment [4-6]. The vehicle industry and other automotive engineering are a perfect area being robotized for the reasons mentioned above. Axiomatically robot's specifications depend on their applications that differ from one purpose to another, like assembly robots [7,8] which are carried out with heavy parts, or PCB manipulators, which need to carry dynamic loads; of course, both previous examples require precise motion. There are many types of industrial robots, and it is used according to the purpose and desired duty [9,10]. The most common type of robot is serial robot manipulators, which are a series of rigid bodies, called links, joined together by means of joints [11], see Figure 2.1. It is economically undesirable to design all the robot manipulators for the same criterion in manufacturing lines of the vehicle industry because the robot's joints and links are subject to different loads in different production lines. It is clear that robot manipulators in assembly lines suffer from more stresses than those in the painting or welding lines. Therefore, there is a need to optimize manipulators links and joints to reach optimum design [12,13]. Another fact should be considered about using robot manipulators in the industry is that the working area or the configuration space of the manipulators may contain static or dynamic obstacles, which leads us to supply the robots with path or trajectory planning. These paths might be a predefined set of points [14] in the Cartesian space in a static environment, or paths continuously change due to the dynamic environment [15]. Still, in both cases, these sets of points should transform from a configuration space to a joint space employing inverse kinematics [16].

1.2 The objective of the study

Due to the importance of optimization algorithms and their wide applications, they could be possible solvers for robotic problems. Also, artificial neural networks were used widely to solve the inverse kinematics of robots, but there is no comparison study about which learning algorithm is the best for inverse problems. Consequently, we summarize the objective of the study as follows:

1- Developing a generic objective function that an optimization algorithm can minimize to find the inverse kinematic solution for any type of robot manipulator with any degree of freedom.

2- Perform a comparison study to find which learning algorithm of neural networks that can be the best choice for inverse kinematic problems.

3- Develop a new educational robotic platform with a unique virtual reality environment and use 3D printing technology to design and implement a new robot arm.

2. GEOMETRIC DESIGN OF THE ROBOT

Six degrees of freedom robot arm was chosen to be designed in this work and containing extra movement for the gripping in the end-effector. The geometric design was fired at first with a product similar to one toy robot existing in the markets, shown on the left of Figure 2.1. On the right of Figure 2.1, there is a modified and larger version of the robot. Both of them were made using a 2 mm thickness of the aluminium sheet. The design was created by CAD file shown in Figure 2.2 and sent to a local company to cut the parts out of a 2 mm aluminium sheet using laser cutter CNC. There are some limitations to this generation; first, the finishing surface is not perfect, and it is easy to scratch the surface accidentally during the assembly of the parts. The laser cutter CNC is expensive, and for each modification on the design, we have to go to the company to cut new parts. This is a tedious process, and even more, only perpendicular sections can be made to form the links of the robot. For example, it is impossible to make a shaft, which is a significant limitation.

3D printing was the next option to manufacture a new generation of the 6DOF robot arm; this process is definitely easy; all that is needed is a 3D printer and filament to start a small factory. There are many filaments with different sets of mechanical properties that make them proper for a wide range of applications. The primary consideration in this work is to release a robot platform with less as much cost as possible.

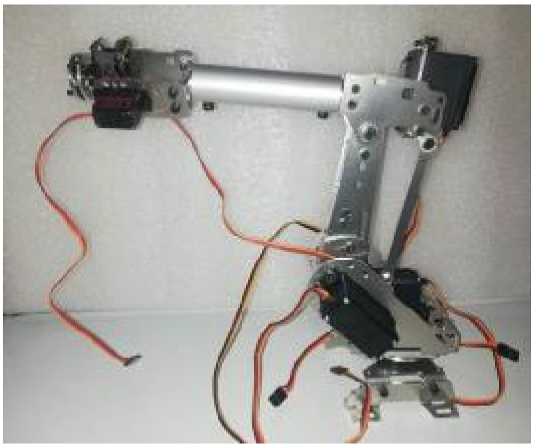
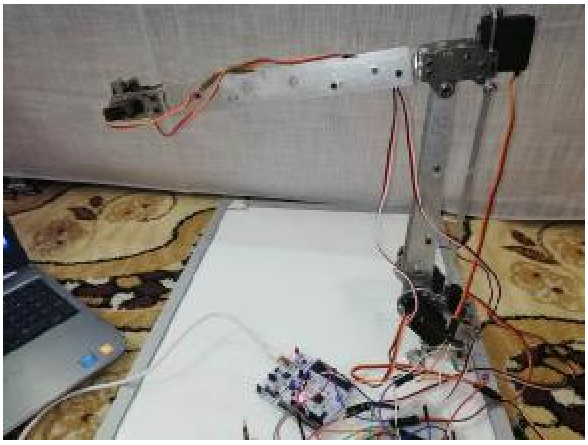



Figure 2.1 Aluminium generations of the 6DOF arm

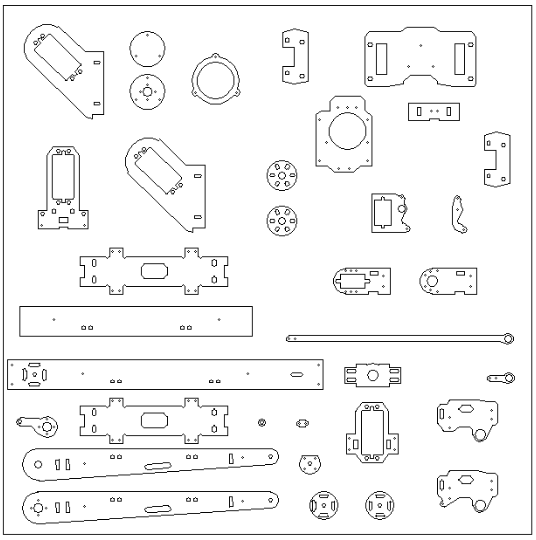


Figure 2.2 Geometric design of the parts of the Aluminium robot arm

The low price was the reason behind choosing PLA and ABS filaments for producing the next generation of the 6DOF arm, as shown in Figure 2.3. PLA is an environment-friendly polymer but has a low thermal resistance, and this is dangerous if, for some reason, the motors get hot during work. ABS has a better thermal resistance and low mechanical strength. The combination of the two materials looked good to print some parts from ABS and other parts from PLA. However, both of them are generally brittle and prone to failure. PLA is easy to print, but it has undesired surface finishing.

On the other hand, ABS has excellent surface finishing, but it is very hard to print. For this reason, and after employing predefined information on the filament, PETG filament had been chosen to print the next and final generation in this study. Figure 2.4 reveals the final version of the 6 DOF robot manipulator that is used in the proposed educational platform.

Figure 2.3 3D printed generation using PLA and ABS filament.

Figure 2.4 3D printed generation using PETG filament.

PETG is quite strong, has good thermal resistance, good surface finishing, and is easy to be printed with a nozzle temperature of about 250 C^o and bed temperature about 70 C^o. This study is for a mechatronic system; rough estimations were adopted because it is out of the scope to study the thermal and strength properties of the used materials. This point could be perfect as a future work for mechanical engineering or material science researchers. There are filaments with super properties like polycarbonate PC and other types that contain nanomaterials in their composition. These filaments had not been employed because of their high prices, considering a low-cost robot in this work.

Another point is that the 3D printing process is slow, not suitable for consumer products, but it is perfect for producing robots. Usually, producer robots that are intended for polishing, grinding, assembling, and painting have low demands on markets. However, one small 3D printer can produce up to four educational robotic arms per day or around 120 pieces per month, which is acceptable for small projects.

3. GENERIC OBJECTIVE FUNCTION

While forward kinematics detects the position and orientation of the end effector from the given set of joint variables, inverse kinematics is the inverse operation. Still, it is more complicated than forward kinematics. The relation between forward and inverse kinematics can be expressed by equation 3.1.

$$\begin{bmatrix} q_1 \\ q_2 \\ \vdots \\ q_n \end{bmatrix} \begin{matrix} \Rightarrow \\ \Leftarrow \end{matrix} \begin{bmatrix} r_{11} & r_{12} & r_{13} & x \\ r_{21} & r_{22} & r_{23} & y \\ r_{31} & r_{32} & r_{33} & z \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3.1}$$

There are many approaches mentioned in other books to deal with this problem, like closed-form and a geometric one. These methods are robot-dependent and differ from one manipulator to another. One can find the solution for the 6DOF manipulator, but hardly or cannot find a theoretical solution for another 5DOF. This section will focus on the solution from the viewpoint of metaheuristics to develop a functional relationship for any robot manipulator. From the perspective of optimization algorithms [17], consider Figure 3.1; for a specific robot configuration, the current position vector of the end-effector can be represented by the distance from the base of the end-effector of the manipulator while the desired position vector represents the task point. Obviously, if the difference between these two vectors is zero, then the tooltip will be in the right position at the task point, and this is the objective function f of the inverse problem

$$f = \|Ci - De\| \tag{3.2}$$

Where *Ci* denotes the instantaneous position vector, and *De* is the desired position vector. In other words, the equation (3.4) is the function that has to be minimized as much as possible, and it is just the distance between the end-effector and task point.

$$f = \sqrt{(x_{Ci} - x_t)^2 + (y_{Ci} - y_t)^2 + (z_{Ci} - z_t)^2} + (R_{Ci} - R_t) \tag{3.3}$$

Where *t* refers to the task point coordinates given for inverse Kinematic problem, R_{ci} is the rotation matrix of the iterative solution provided by the optimization algorithm, andR_t is the desired or given rotation matrix of the end-effector.

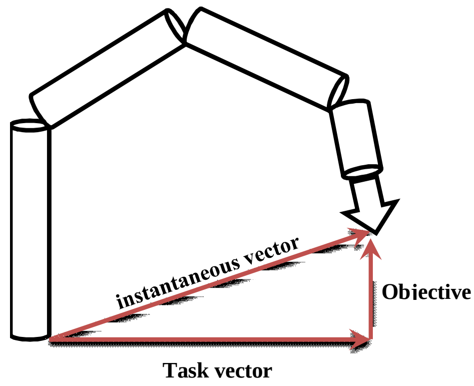


Figure 3.1 Representation of the objective function for inverse Kinematic problem

If the first term of the equation (3.3) has been used alone as an objective function, we may get the end-effector in the task point but with many choices of orientations. Figure 3.2 reveals an example of two orientation options for the same task point, and manipulators with a high degree of freedom have even an infinite number of orientations.

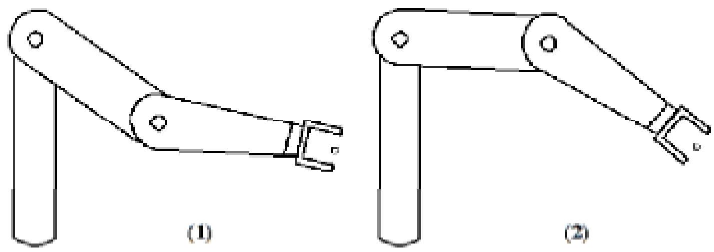


Figure 3.2 Different orientations for the same position

Once again, the inverse Kinematic solution obtained from optimization algorithms needs time measured in minutes (at least one or half a minute). This is inconvenient if we need a quick response in part of a second. The inverse kinematic problem here is a minimization problem where equation 3.3 should be driven to zero. The best performance was found by using particle swarm optimization. However, it is not global since PSO can’t return a solution with a cost function value less than 59 even with a high number of iteration and populations. In other words, PSO, like other metaheuristics, falls into local minima and can’t escape from that to the global point in the search space. However, the provided solution by PSO is reliable and acceptable, but the big difference will be on orientation value. For each independent run, PSO will return a similar position vector and totally different configuration or, say, different orientation, see Figure 3.3. We can accept falling in local minima and choose the proper configuration for a point in space by setting constraints during the iterative process of the

PSO. However, it is worth considering that the developed objective function in this section can solve the inverse kinematics of any type of serial robot arm with any degree of freedom.

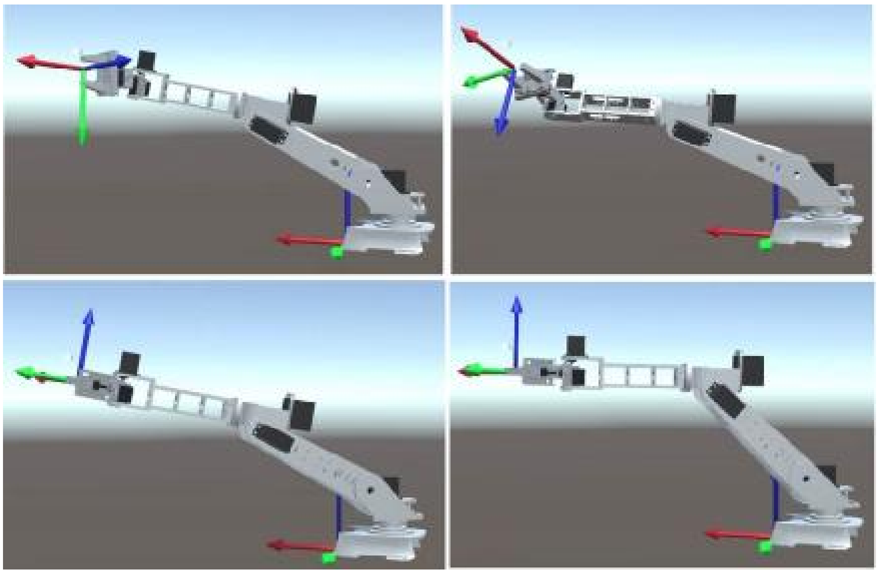


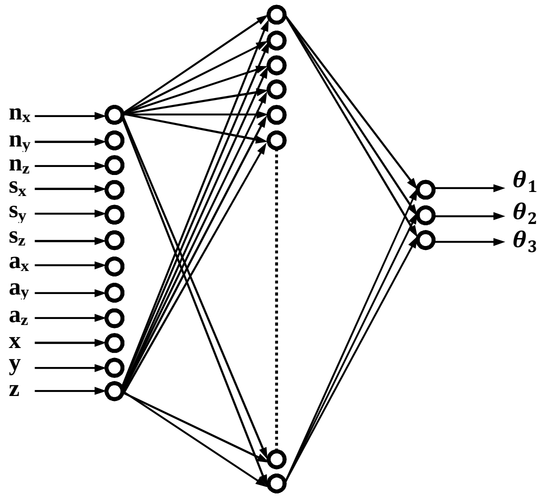
Figure 3.3 Different configurations for each independent run for the PSO algorithm

4. ANN FOR INVERSE KINEMATICS

In this section, we will focus on the solution from the viewpoint of neural networks to develop a functional relationship for any robot manipulator. While forward equations are a straightforward process, we will rely on these equations to establish the set of training data for the inverse problem. From forward Kinematic equations, a data set of 2000 input/output training data is generated to learn the proposed topology of neural networks how to solve the inverse position problem of the robot using information out of the training data. Figure 4.1 reveals the topology of the artificial neural network used in this study, where it has 12 inputs, one hidden layer, and three outputs. The input layer is the element of the homogeneous transformation matrix that relates the end-effector to the base frame. In this section, we have used one hidden layer with several neurons n=10 and n=100. The output layer consists of three neurons that represent the three joint angles of the robot arm.

4.1 RESULTS AND DISCUSSION

Three learning algorithms are available for MATLAB neural network toolbox, and we have employed them and compared the obtained results in cases of n=10 and n=100. The proposed neural network has been learned using three learning algorithms; Levenberg-Marquardt algorithm LM, Bayesian Regularization algorithm BR, and Scaled Conjugate Gradient algorithm SCG.



n=10, 100

Figure 4.1 ANN architecture of RRR robot manipulator

Each of these algorithms has its own advantage and disadvantage that should be explained in this study. Figure 4. to Figure shows the performance of Levenberg-Marquardt, Bayesian Regularization, and Scaled Conjugate Gradient, respectively, on the learning process in the case of several neurons equal to 10. The Levenberg-Marquardt algorithm needs less time than Bayesian Regularization and more time than Scaled Conjugate Gradient, but it needs more memory. Also, LM is more accurate than SCG and less accurate than BR. By repeating the same comparison in the case of n=100, see Figure to Figure 4.13, it is easy to configure the huge reduction in mean square error during the test compare with the case when n=10. A data set of 2000 items is used for the learning process where 70% of the items are used for training while 30% have been used for the test. Axiomatically, an increasing number of neurons in the hidden layer can lead to the best results with less error. Still, here we have to evaluate the performance of the three learning algorithms. Table 4.1 reveals the elapsed time to learn the network how to solve inverse kinematics of the robot where BR needs more time than LM while SCG has taken a very short time. Back to Figure to Figure , increasing the number of neurons to 100 has led to a dramatic drop in mean square error MSE in the case of LM and BR learning algorithms. However, the SCG algorithm still cannot learn the network solving inverse Kinematic.

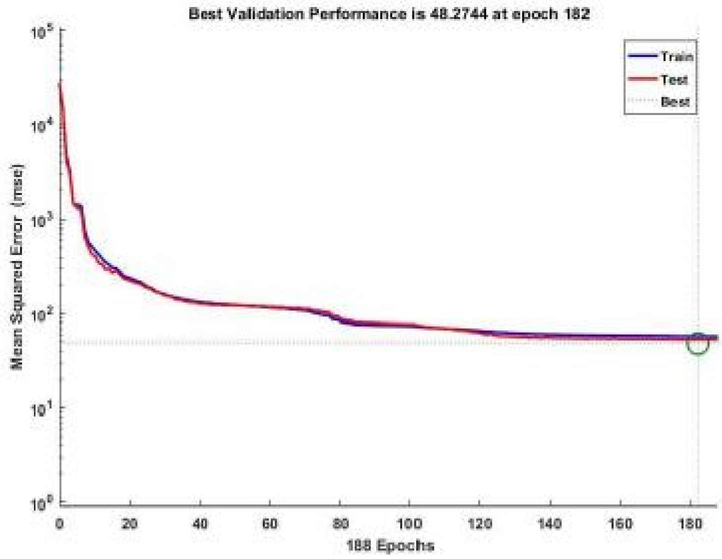


Figure 4.2 Mean square error for Levenberg-Marquardt algorithm with n=10

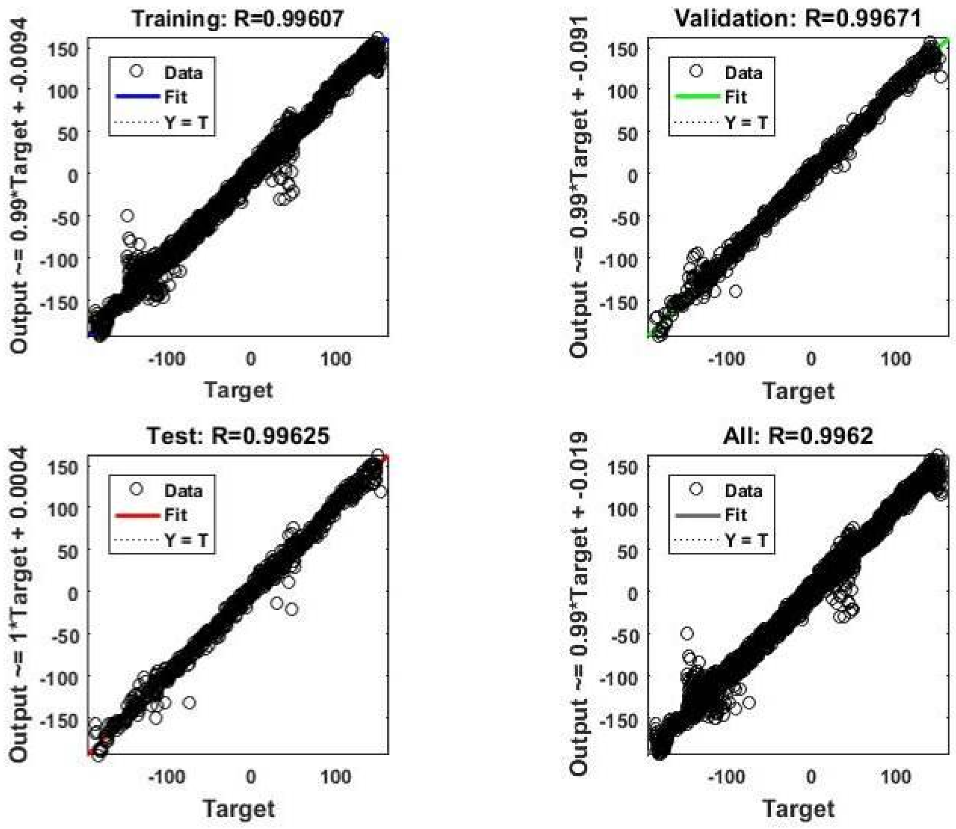


Figure 4.3 Regression of Levenberg-Marquardt algorithm with n=10

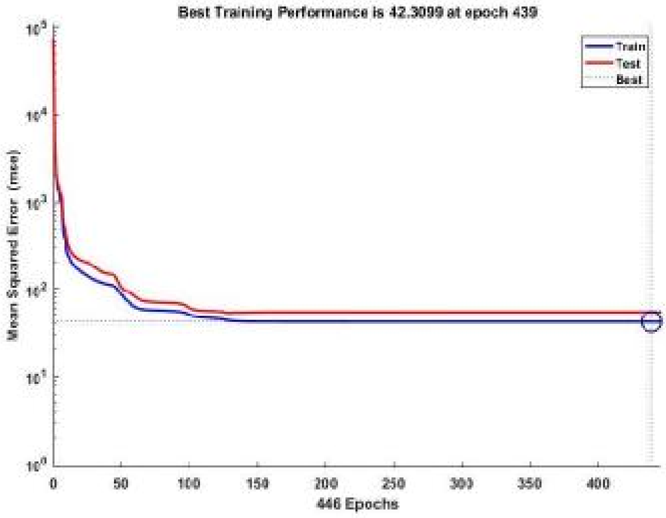


Figure 4.4 Mean square error for Bayesian Regularization algorithm with n=10

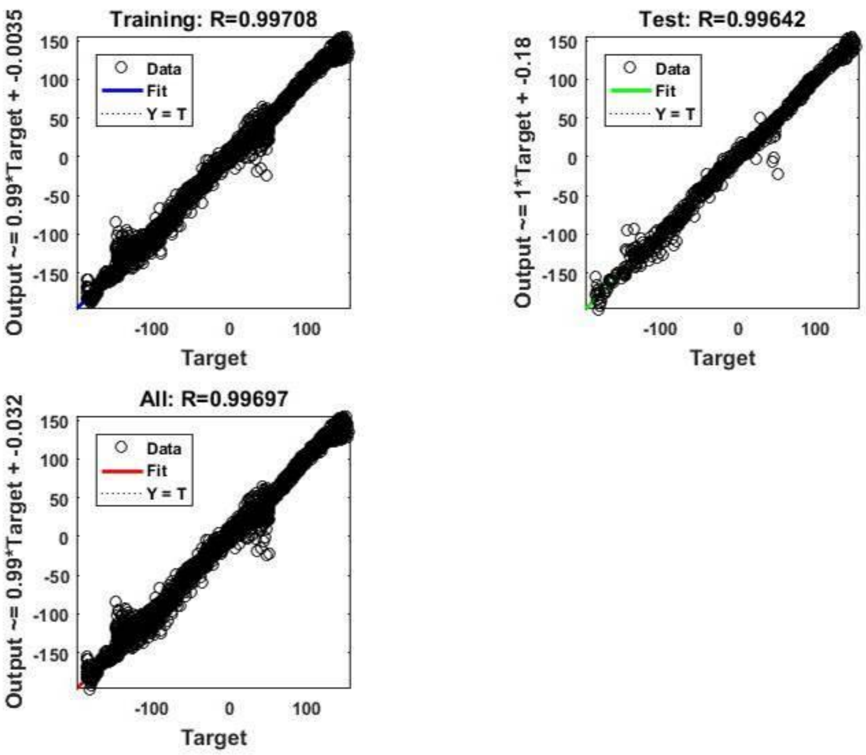


Figure 4.5 Regression of Bayesian Regularization algorithm with n=10

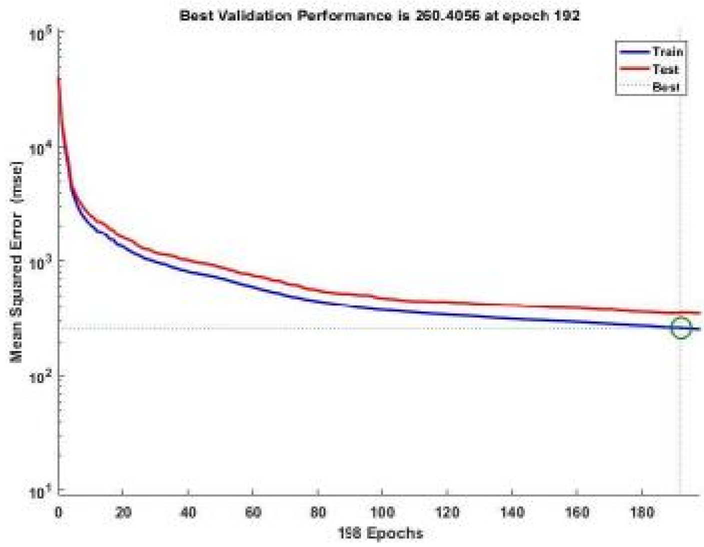


Figure 4.6 Mean square error for Scaled Conjugate Gradient algorithm with n=10

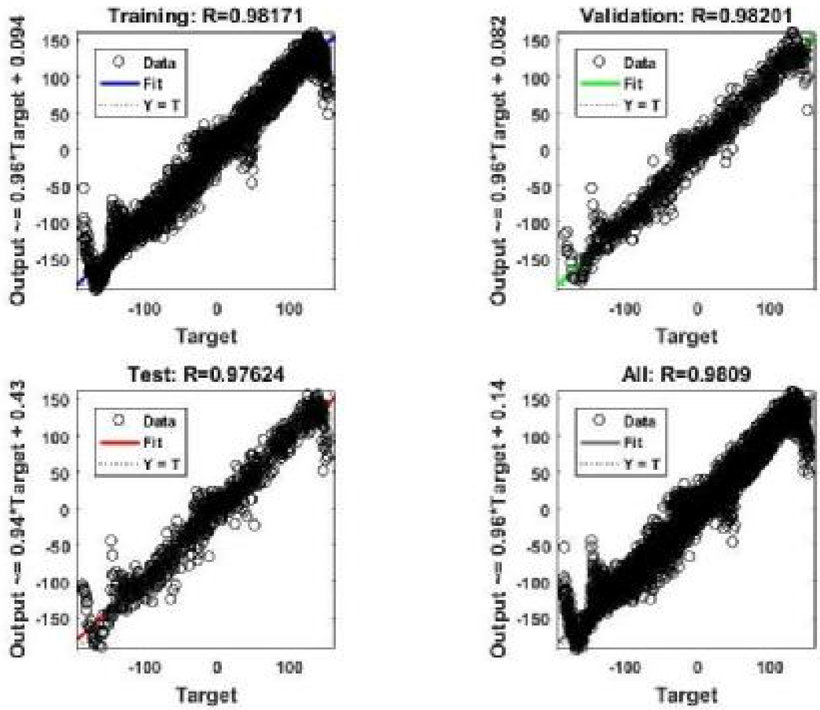


Figure 4.7 Regression of the Scaled Conjugate Gradient algorithm with n=10

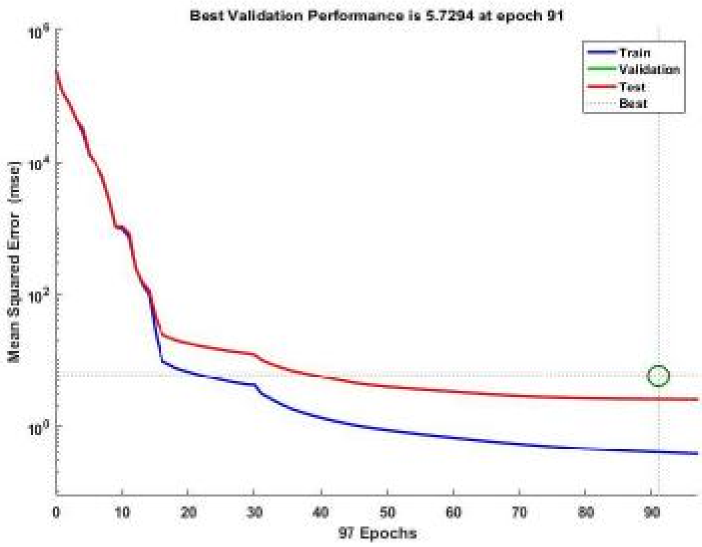


Figure 4.8 Mean square error for Levenberg-Marquardt algorithm with n=100.

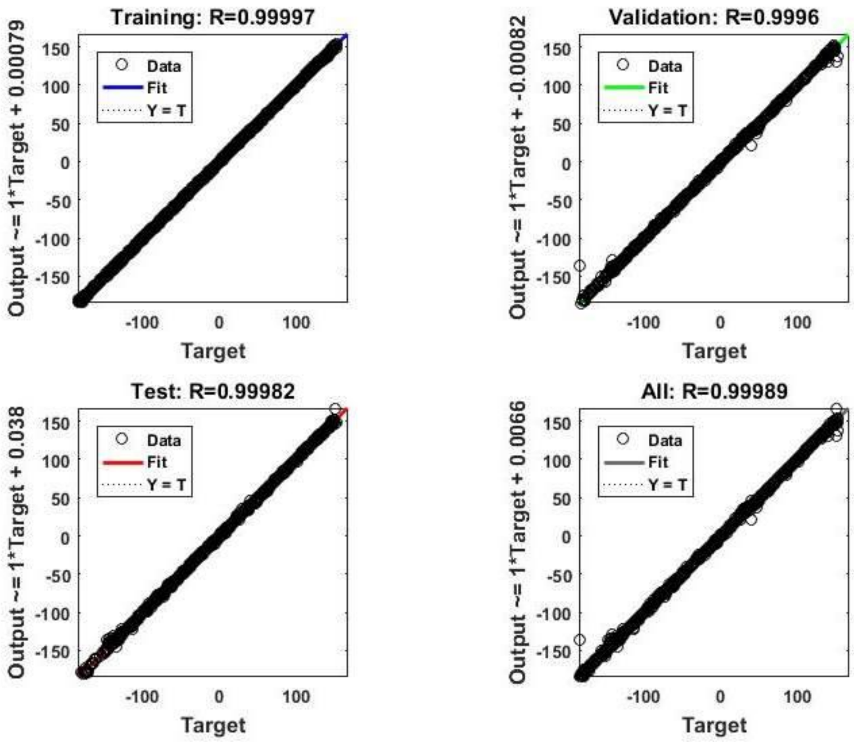


Figure 4.9 Regression of the Levenberg-Marquardt algorithm with n=100

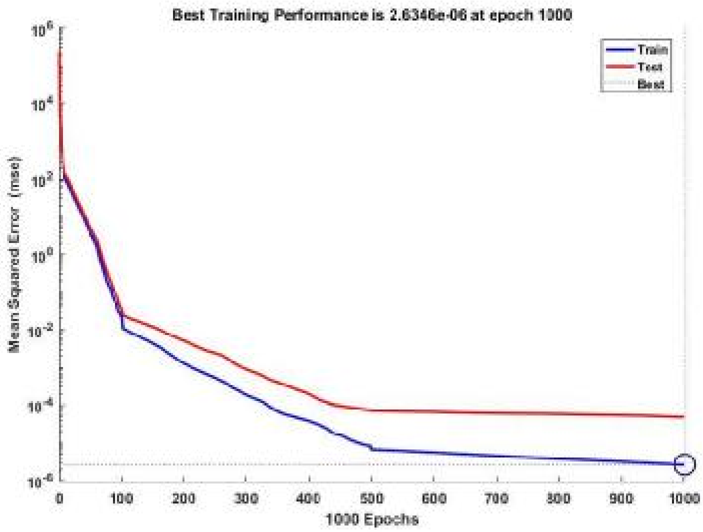


Figure 4.10 Mean square error for Bayesian Regularization algorithm with n=100.

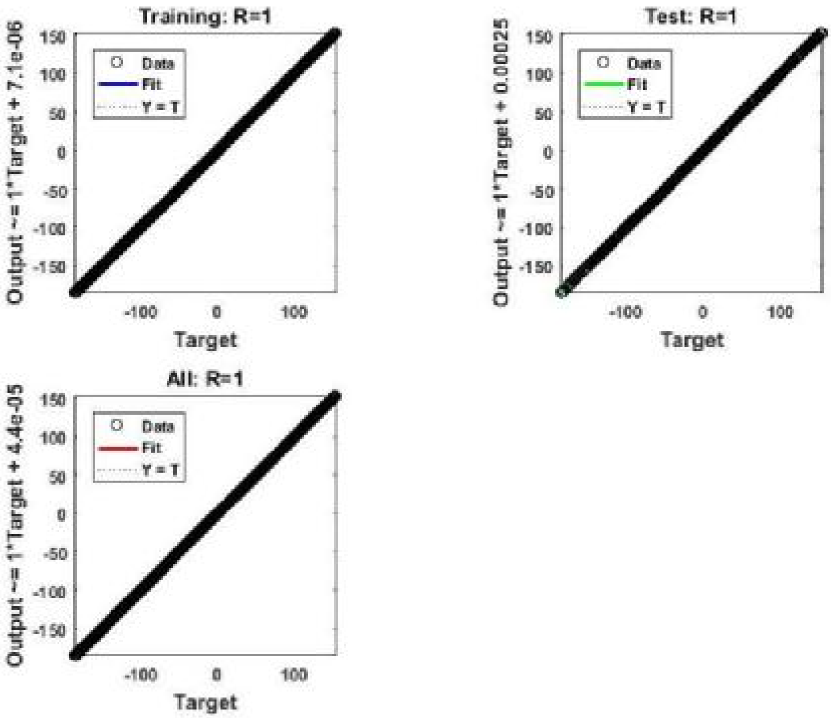


Figure 4.11 Regression of the Bayesian Regularization algorithm with n=100

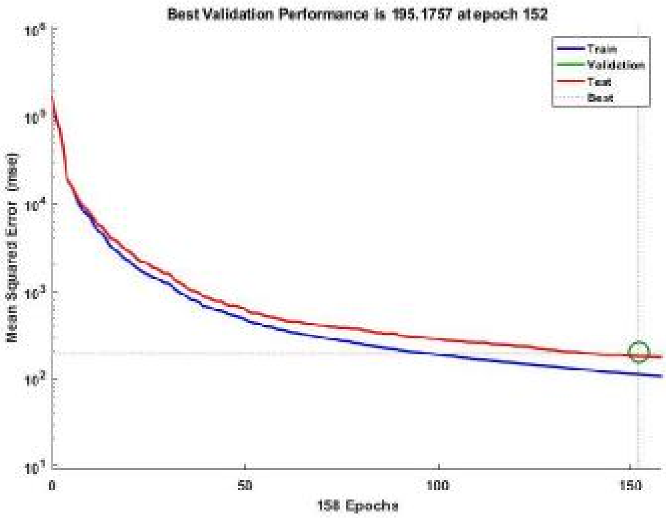


Figure 4.12 Mean square error for Scaled Conjugate Gradient algorithm with n=100.

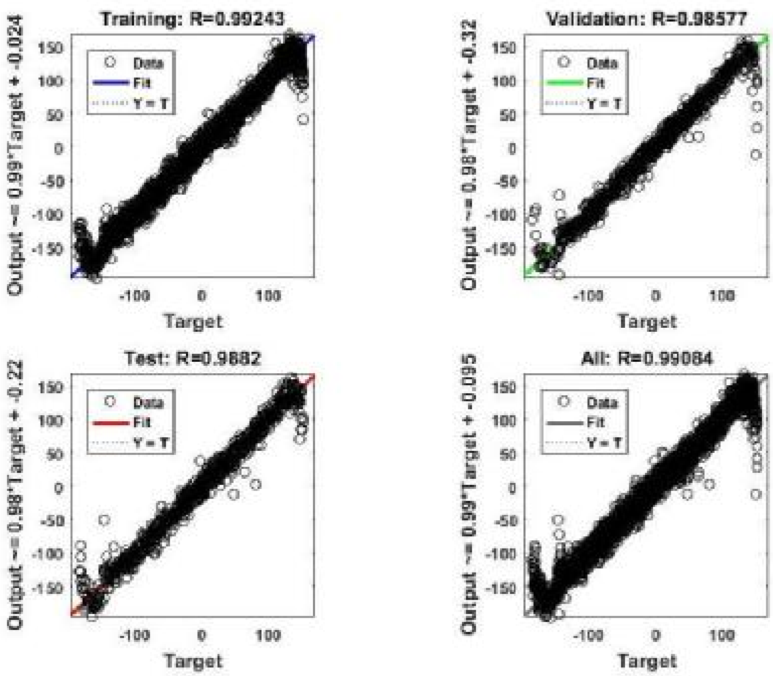


Figure 4.13 Regression of the Scaled Conjugate Gradient algorithm with n=100

Figure to Figure show the weak performance of the SCG algorithm on this kind of nonlinear problem. Table 4.1 shows the time consumption by the three algorithms while Table 4.2 demonstrates MSE induced by using the three mentioned learning algorithms on the proposed problem.

Table 4.1 Time consuming of Levenberg-Marquardt, Bayesian Regularization, and Scaled Conjugate Gradient learning algorithms

learning algorithm	time (sec)	
	n=10	n=100
Levenberg-Marquardt	4	171
Bayesian Regularization	8	1840
Scaled Conjugate Gradient	0.01	1

Table 4.2 Mean square error of using Levenberg-Marquardt, Bayesian Regularization, and Scaled Conjugate Gradient learning algorithms

learning algorithm	Mean square error MSE	
	n=10	n=100
Levenberg-Marquardt	48.2744	5.7294
Bayesian Regularization	42.3099	2.6346E-06
Scaled Conjugate Gradient	260.4056	195.1757

A neural network is a powerful tool for nonlinear problems like inverse kinematic problems. The neural network represents a function between inputs and outputs represented by distributed weights on connections among neurons. In this section, we have studied the effect of using three different learning algorithms, namely, Levenberg-Marquardt algorithm, Bayesian Regularization algorithm, and Scaled Conjugate Gradient algorithm, to learn a neural network how to solve inverse Kinematic of a three revolute joints robot manipulator. The topology of the network consists of one input layer with twelve neurons, one hidden layer with a number of neurons n=10, 100, and one output layer with three neurons. The twelve neurons in the input layer represent the given element of the homogeneous transformation matrix, while the three neurons in the output layer represent the desired joint angles of the robot. During the study, we found that the Scaled Conjugate Gradient learning algorithm was unable to learn the proposed problem's network, even with a high number of neurons in the hidden layer. Bayesian Regularization learning algorithm returned the best results, but with elapsed time greater than what is required for the Levenberg-Marquardt algorithm.

5. EDUCATIONAL ROBOTIC PLATFORM

Figure 5.1 illustrates the educational robotic platform with a 3D printed arm and Unity 3D-based host application. It consists of 6 DOF robot manipulators with gripping ability, a microcontroller device as the hardware of the robot controller, and a host application fixed on a PC.

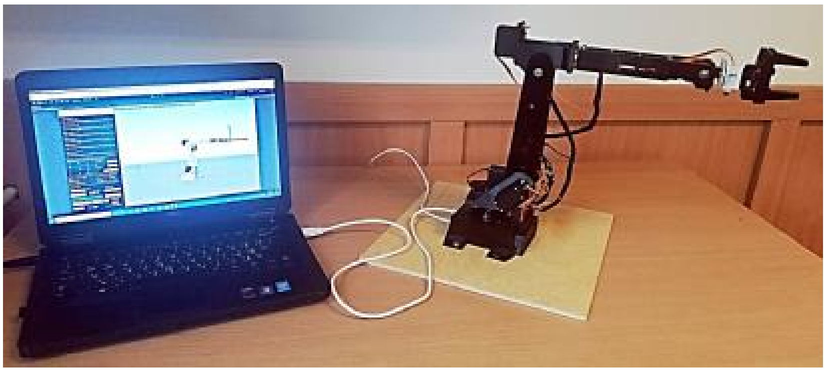


Figure 5.1 Improved educational robotic platform

5.1 THE CONTROLLER

The host application sends discrete data in ranges of the servo motors, which is [0,180]. In other meaning, servo motors move physically, between 0 and 180, but for mathematical purposes, we considered the range [-90,90] as an input by the final user. The above-mentioned mapping occurs on the host side before sending to the target device, as shown in Figure 5.2.

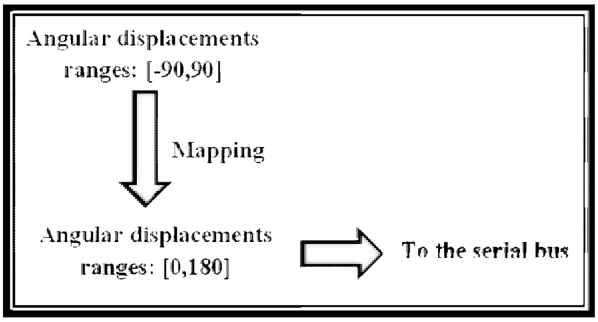


Figure 5.2 Mapping mathematical to actual ranges of the servos

On the other side, the target device, which is STM32 microcontroller, is simultaneously controlling six servo motors by sending frequent pluses on their control line.as revealed in Figure 5.3. According to the clock configuration of the microcontroller, the minimum servo position occurs at 1 ms, which is equivalent to 500 on the timer capture/compare register. The maximum servo position for this type of motor occurs by setting the duty cycle of 2 ms, which corresponds to 2400 on the capture/compare register of the timers. In other words, the range of motion of the motors on the target device is [500, 2400].

Thus, a novel equation is developed in this study to map the range of motion from [0^o,180^o] to [500,2400] as follows:

$$D = 500 + dis\frac{160}{15} \tag{5.1}$$

Where D is the angular displacement of the motor on the range [500,2400] while *this* is the displacement on [0,180], the error percentage is very low using equation (5.1). For example, 180^o can be transformed to 2420 while it should be 2400.

5.2 HOST APPLICATION

Unity 3D and C# .NET 2019 were used to develop the host application that controls the robot. Figure 5.4 reveals the host application that has to control the physical robot on the real-world side, where it explains the forward kinematic section. The application uses the forward kinematic equations to return the position and orientation of the end-effector relative to the base of the robot.

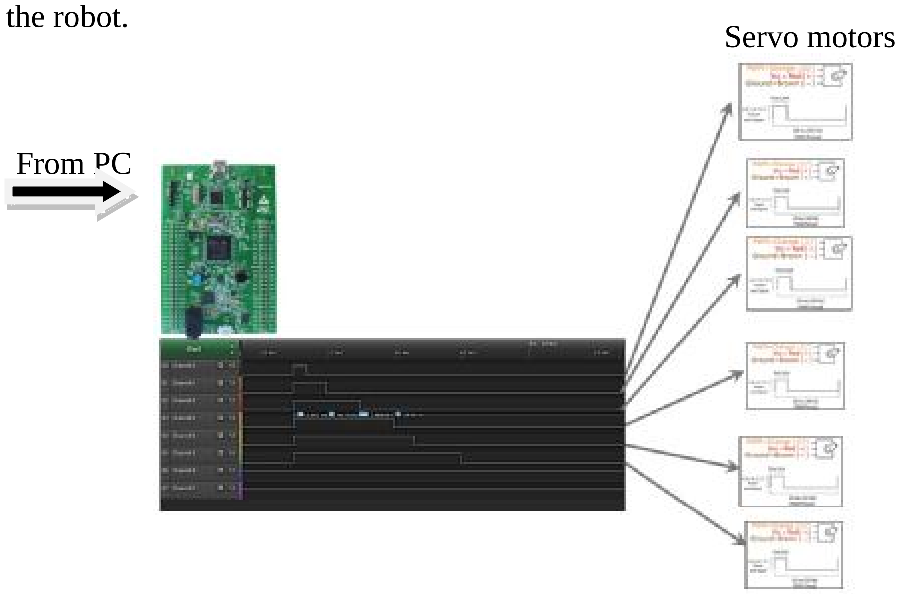


Figure 5.3 Data serialization from the host application to motors

Everything in this application was built from scratch without any tools only by using C# libraries and Unity 3D game engine. It is worth saying that Unity 3D is originally founded for video games developments. Thus, our application is a kind of game environment with a connection to the real world via a serial bus to a microcontroller that controls a group of servo motors. In other words, the proposed robotic platform is a virtual reality application synchronizing a virtual environment on a computer with real-world equipment.

Figure 5.4 illustrates the forward kinematic application on the proposed system.

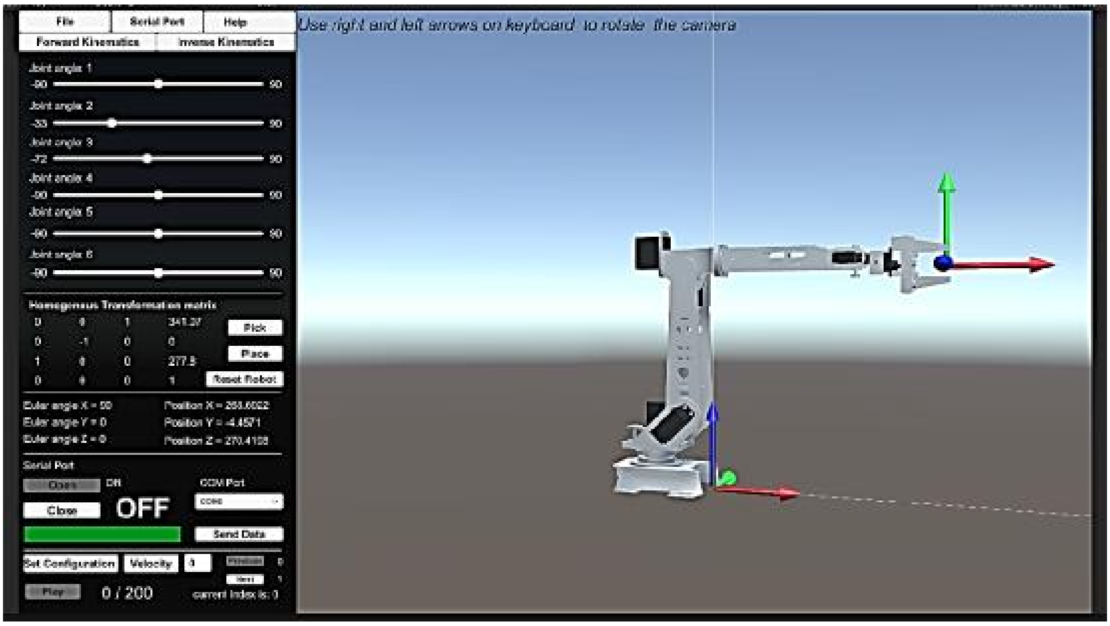


Figure 5.4 Host application written using Unity 3D and C# .NET

Figure 5.5 shows the inverse kinematic section in the application where the inputs are the position and orientation of the end-effector. At the same time, the outputs are a set of joint angular displacements. Particle swarm optimization takes the inputs and, after several trials, returns the outputs, and the mechanism is described in section 3. The application can record different motions, save them, and reload them again in different sessions.

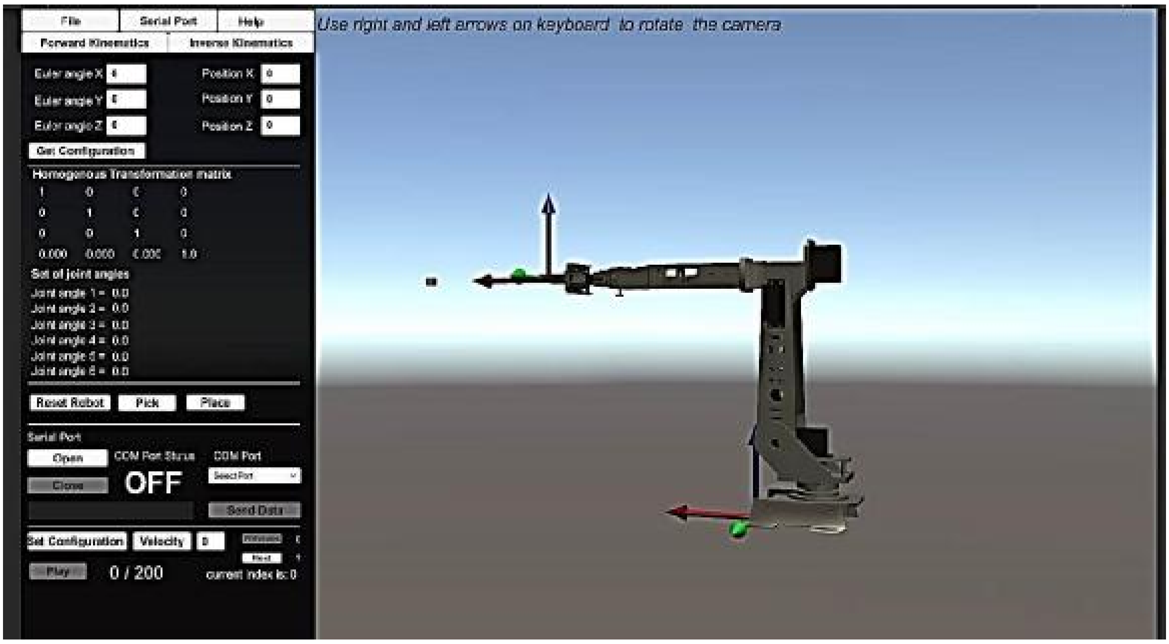


Figure 5.5 Inverse kinematic section

6. NEW SCIENTIFIC RESULTS – THESES

T1. Building a robust controller for the robot manipulator to control its movements and configurations. The unique controller consists of two distinguished subsystems, one of the host and another on the target device. The first subsystem, which lies on the host machine, takes user commands as inputs and gives digital values for the joint angles of the virtual model as outputs. The second subsystem, which lies on the microcontroller device, takes the digital values of the joint angles of the virtual model as inputs and gives electrical signals (PWM's) as outputs to the corresponding servo motors. Serial communications via USART play the role of a bridge between the two systems. In this work, the advantages of the ARM-based microcontroller STM32F407 discovery are used by setting six different timer channels; each one corresponds to a servo motor. Creating and Developing a virtual reality application using a game engine and C#.NET framework to handle user commands through a graphical user interface. Game engines are widely used to develop video games, and this work has been employed them to build a simulation environment for the robot. This application takes the user commands as inputs and sends the outputs to the microcontroller over a bridge of serial communications. The advantage of this application is that it offers a visualized interface to the user to see and decide where to move the robot within a constrained workspace with obstacles. This application is perfect for remote-controlling in a predefined workspace that can be dangerous or poisonous. Users can only deal with a virtual work area that is similar to the real one and control the manipulator remotely.

T2. Developing a new artificial neural network to solve the inverse kinematic problem of the robot. ANN consists of a single layer with 100 neurons is developed to take the Cartesian position of the end-effector of the robot as input and return the corresponding joint angles as outputs. This technique provides the controller with the ability to give the robot a rapid response with an acceptable error in positioning the end-effector.

T3. Developing generic objective function that can be used to solve the inverse kinematic problem using optimization algorithms. There are a variety of meeting heuristics with different specifications and abilities to solve optimization problems. In this work, a general-purpose objective function is developed to be optimized by any metaheuristic or heuristic algorithm. The optimization problem is minimizing the error in positioning the end-effector by setting appropriate or optimal joint angles for the manipulator. The developed objective function is differentiable, and the global value is known as zero. Thus, we have found that classical optimizers like hill-climbing are better than other met heuristics algorithms to find optimal joint angles related to a known end-effector position. The inverse kinematic problem in robotics is very complicated and tedious; some robots with 7 DOF or more impossible to find inverse solutions for them using traditional and analytical methods. By developing this objective function, the solution for the inverse kinematic problem becomes very easy and straightforward, even for those robots with a high degree of freedom. However, the optimization method is more time-consuming and accurate than using ANN. Thus, an optimization method was added to the controller in case of slow motion, and precise movements were required.

7. LIST OF PUBLICATIONS RELATED TO THE TOPIC OF THE RESEARCH FIELD

(1) Hazim, Nasir Ghafil ; Dr. Jármai, Károly Optimáló algoritmusok robotok inverz kinematikájához MATLAB forráskóddal GÉP 72 : 1-2 pp. 71-74. , 4 p. (2021)

(2) Hazim, Nasir Ghafil ; Karoly, Jarmai Optimization Algorithms for Inverse Kinematics of Robots with MATLAB Source Code LECTURE NOTES IN MECHANICAL ENGINEERING 22 pp. 468-477. , 10 p. (2021)

(3) Ghafil, Hazim Nasir ; Jármai, Károly Dynamic differential annealed optimization: A new metaheuristic optimization algorithm for engineering applications APPLIED SOFT COMPUTING 93 Paper: 106392 (2020)

(4) Hazim, Nasir Ghafil ; Jármai, Károly IDEGHÁLÓK ROBOTOK INVERZ KINEMATIKAI VIZSGÁLATÁHOZ: ÁTTEKINTÉS ÉS ALKALMAZÁS GÉP 71 : 2 pp. 5-10. , 6 p. (2020)

(5) Hazim, Nasir Ghafil ; Károly, Jármai Optimization for Robot Modeling with MATLAB Basel, Switzerland: Springer Nature Switzerland AG (2020), 220 p. DOI ISBN: 9783030404093 ISBN: 3030404099 ISBN: 9783030404109

(6) Hazim, Nasir GHAFIL; S., ALSAMIA; Károly, JÁRMAI Novel Levy flight stochastic optimization algorithm for various engineering problems In Iványi, Péter Abstract book for the 16th MIKLÓS IVÁNYI INTERNATIONAL PhD & DLA SYMPOSIUM (2020) p. 124, 1 p.

(7) Alsarayefi, Saad; Jálics, Károly; Ghafil, Hazim Nasir Optimization of the fiber size for a fiber glass epoxy composite DESIGN OF MACHINES AND STRUCTURES 9 : 1 pp. 5-12. , 8 p. (2019) REAL Complete document

(8) Ghafil, Hazim Nasir; Jármai, Károly Kinematic-based structural optimization of robots POLLACK PERIODICA: AN INTERNATIONAL JOURNAL FOR ENGINEERING AND INFORMATION SCIENCES 14 : 3 pp. 213-222. , 10 p. (2019)

(9) Hazim, Nasir ; Jármai, Károly A részecske csoport és a mesterséges méhcsalád módszerek összehasonlító vizsgálata GÉP 70 : 2 pp. 3-8. , 6 p. (2019).

(10) Hazim, Nasir Ghafil ; Kovács, László ; Károly, Jármai Investigating three learning algorithms of a neural networks during inverse kinematics of robots In: Szita Tóthné, Klára; Jármai, Károly; Voith, Katalin (ed.) Solutions for Sustainable Development: Proceedings of the 1st International Conference on Engineering Solutions for Sustainable Development, (ICESSD 2019) London, United Kingdom / England: CRC Press (2019) 394 p. pp. 33-40. , 8 p.

(11) Hazim, Nasir Ghafil; Károly, JármaiA novel hybrid algorithm with application in the kinematic-based structural optimization of robots In: Proceedings of the 13th World Congress on Structural and Multidisciplinary Optimization (WCSMO 13) International Society for Structural and Multidisciplinary Optimization (ISSMO), (2019) pp. 206-206. 1 p.

(12) Hazim, Nasir Ghafil ; Károly, Jármai On Performance of ED7500 and Lab-Volt 5150 5DOF Manipulator: Calibration and Analysis In: Kékesi, Tamás (ed.) MultiScience – XXXIII. microCAD International Multidisciplinary Scientific Conference University of Miskolc, Hungary: University of Miskolc (2019) pp. 1-9. Paper: D2-4, 9 p.

(13) HN, Ghafil; K., Jármai Optimum dynamic analysis of a robot arm using flower pollination algorithm In: Peter, Platko; Mohamad, Al Ali (eds.) Advances and Trends in Engineering Sciences and Technologies III. London, UK / England: CRC Press – Taylor and Francis Group, (2019) pp. 81-88. , 8 p.

(14) Mahmood Badr Ali, Doaa; Nasir Ghafil, Hazim; Károly, JármaiOptimizing Heuristic Graph Formation with Application in Kinematic Synthesis of a Robot Arm with Revolute Joints JOURNAL OF ENGINEERING AND APPLIED SCIENCES 14: 6 pp. 1976-1948. Paper: 14 (2019)

(15) Ghafil, Hazim; Jármai, Károly Optimal stress-strain analysis of a robot arm through finite-element and simulated annealing algorithm In: Fülöp, Attila; Iványi, Péter (eds.) Abstract book for the 14th Miklós Iványi International PhD & DLA Symposium: Architectural, Engineering and Information Sciences Pécs, Hungary: Pollack Press, (2018) p. 70 Paper: 26

(16) Ghafil, Hazim ; Jármai, Károly Kinematic based structural optimization of robots In: Fülöp, Attila; Iványi, Péter (eds.) Abstract book for the 14th Miklós Iványi International PhD & DLA Symposium: Architectural, Engineering and Information Sciences Pécs, Hungary: Pollack Press, (2018) p. 69 Paper: 25

(17) Ghafil, Hazim; Shaymaa, Mahmood Badr MATLAB FOR ENGINEERS- BASICS AND APPLICATIONS WORLD JOURNAL OF ENGINEERING RESEARCH AND TECHNOLOGY 4: 1 p. 1 (2018).

(18) Ghafil, HN; Jármai, K Research and application of industrial robot manipulators in vehicle and automotive engineering, a survey, Lecture Notes in Mechanical Engineering, pp. 611-623. Springer International Publishing AG, ISBN 978-3-319-75677-6 (eBook) https://doi.org/10.1007/978-3-319-75677-6.

(19) Ghafil, HN; Jármai, K Comparative study of particle swarm optimization and artificial bee colony algorithms In: Kékesi, Tamás (ed.) Multiscience XXXII. MicroCAD International Multidisciplinary Scientific Conference University of Miskolc, Hungary: University of Miskolc (2018) Paper: D1_Hazim_Nasir_Ghafil.pdf, 6 p.

(20) Hazim, Nasir Ghafil ; Jármai, Károly Ipari robot és manipulátorok kutatása és alkalmazása jármű-és autóipari mérnöki területeken, áttekintés GÉP 69 : 2 pp. 36-43. , 8 p. (2018).

(21) H N, Ghafil; K, Jármai Trajectory planning of a robot arm using flower pollination algorithm In: Peter, PLATKO; Mohamad, AL ALI; Štefan, KUŠNÍR (eds.) 3rd International Conference on Engineering Sciences and Technologies: ESAT 2018 Košice, Szlovákia: Technical University of Kosice, (2018) Paper: 023 – Ghafil, Jarmai.pdf , 4 p.

(22) Ghafil, Hazim Nasir Inverse Acceleration Solution for Robot Manipulators using Harmony Search Algorithm INTERNATIONAL JOURNAL OF COMPUTER APPLICATIONS 6: 114 pp. 1-7, 7 p. (2016).

(23) Nasir Ghafil, Hazim Vibration Analysis of a Generator Anti-Vibration Rubber Mounts Journal of Computer Science & Computational Mathematics [electronic only] 6: 4 pp. 105-111, 7 p. (2016).

(24) HN Ghafil, AH Mohammed, NH Hadi A Virtual Reality Environment for 5-DOF Robot Manipulator based on XNA Framework INTERNATIONAL JOURNAL OF COMPUTER APPLICATIONS 113 : 3 pp. 33-37. 5 p. (2015).

(25) HN Ghafil, AH Mohammed, NH Hadi. "Optimum Path Planning and Performance Analysis of a Robot Manipulator.". Thesis (2013), Alnahrain university Baghdad- Iraq.

(26) Nasir Ghafil H, Alsamia S, Jármai K (2021) Fertilization optimization algorithm on CEC2015 and large scale problems. Pollack Periodica. DOI: https://doi.org/10.1556/606.2021.00343

(27) Alsamia S, Ibrahim DS, Ghafil HN (2021) Optimization of drilling performance using various metaheuristics. Pollack Periodica, DOI: https://doi.org/10.1556/606.2021.00307

(28) Hazim Albedran et al, "Interpolated spline method for a thermal distribution of a pipe with a turbulent heat flow", Multidisciplinary Sciences Vol. 11 No. 5 (2021). URL: https://ojs.uni-miskolc.hu/index.php/multi/article/view/883/659.

(29) Ali Habeeb Askar, Hazim Albedran, et al, "A new method to predict temperature distribution on a tube at constant heat flux". Multidisciplinary Sciences Vol. 11 No. 5 (2021). URL: https://ojs.uni-iskolc.hu/index.php/multi/article/view/884/660

8. LITERATURE CITED IN THE THESES BOOKLET

[1] Daneshmand M, Bilici O, Bolotnikova A, Anbarjafari G (2017) Medical robots with potential applications in participatory and opportunistic remote sensing: A review. Robotics and Autonomous Systems 95:160-180.

[2] Wang H, Chen W, Lai Y, He T (2015) Trajectory planning of tokamak flexible in-vessel inspection robot. Fusion Engineering and Design 98:1678-1682

[3] Spolaôr N, Benitti FBV (2017) Robotics applications grounded in learning theories on tertiary education: A systematic review. Computers & Education 112:97-107

[4] Gopinath V, Johansen K (2016) Risk assessment process for collaborative assembly–a job safety analysis approach. Procedia CIRP 44:199-203

[5] Gopinath V, Ore F, Johansen K (2017) Safe assembly cell layout through risk assessment–an application with hand guided industrial robot. Procedia CIRP 63:430-435

[6] Michalos G, Makris S, Spiliotopoulos J, Misios I, Tsarouchi P, Chryssolouris G (2014) ROBO-PARTNER: Seamless human-robot cooperation for intelligent, flexible and safe operations in the assembly factories of the future. Procedia CIRP 23:71-76

[7] Kramberger A, Gams A, Nemec B, Chrysostomou D, Madsen O, Ude A (2017) Generalization of orientation trajectories and force-torque profiles for robotic assembly. Robotics and Autonomous Systems 98:333-346

[8] Makris S, Tsarouchi P, Matthaiakis A-S, Athanasatos A, Chatzigeorgiou X, Stefos M, Giavridis K, Aivaliotis S (2017) Dual arm robot in cooperation with humans for flexible assembly. CIRP Annals 66 (1):13-16

[9] Spong MW (2005) Seth. Hutchinson, and M. Vidyasagar. Robot Modeling and Control.

[10] Craig JJ (2009) Introduction to robotics: mechanics and control, 3/E. Pearson Education India,

[11] Craig JJ (2009) Introduction to robotics: mechanics and control, 3/E. Pearson Education India,

[12] Ghafil HN, Mohammed AH, Hadi NH (2015) A virtual reality environment for 5-DOF robot manipulator based on XNA framework. International Journal of Computer Applications 113 (3)

[13] Kivelä T, Mattila J, Puura J (2017) A generic method to optimize a redundant serial robotic manipulator's structure. Automation in Construction 81:172-179

[14] Hassan A, Abomoharam M (2017) Modeling and design optimization of a robot gripper mechanism. Robotics and Computer-Integrated Manufacturing 46:94-103

[15] Mathew R, Hiremath SS (2016) Trajectory tracking and control of differential drive robot for predefined regular geometrical path. Procedia Technology 25:1273-1280

[16] Tuncer A, Yildirim M (2012) Dynamic path planning of mobile robots with improved genetic algorithm. Computers & Electrical Engineering 38 (6):1564-1572

[17] Iliukhin V, Mitkovskii K, Bizyanova D, Akopyan A (2017) The modeling of inverse kinematics for 5 DOF manipulator. Procedia Engineering 176:498-505

[18] Ghafil HN, Jármai K Optimization Algorithms for Inverse Kinematics of Robots with MATLAB Source Code. In: Vehicle and Automotive Engineering, 2020. Springer Singapore, pp 468-477. https://doi.org/10.1007/978-981-15-9529-5_41