Journal of Engg. Research Vol. 9 No. (3B) September 2021 pp. 235-247 DOI : 10.36909/jer.v9i3B.8807

End effector position calculation with the ANN for tapping machine

Gullu Akkas*, Ihsan Korkut** and Murat Tolga Ozkan***

**Kayseri University, Junior Technical College, Mechanical and Metal Technologies Department, Kayseri, 38000, TURKEY*
***Gazi University, Faculty of Technology Department of Manufacturing Engineering, Ankara, 06500, TURKEY*
****Gazi University, Faculty of Technology Department of Industrial Design Engineering, Ankara, 06500, TURKEY*
**Corresponding Author: gullu.akkas@kayseri.edu.tr*

Submitted: 14/10/2019
Revised: 03/12/2020
Accepted: 08/12/2020

ABSTRACT

Nowadays, manufacturers give importance to the production of machines that allow for faster production, reduce labor costs, and minimize operation errors to meet the increasing demand. The search for such machines leads the manufacturing sector to automation. In this study, an automation-supported tapping machine prototype was manufactured. Kinematic equations were used for determining the location of the end effector in Cartesian space, whereas inverse kinematic equations were used for angular positions in joint space relative to positions in Cartesian space. Based on the results of the kinematic equations, the data obtained in certain positions were taught to the system through ANN. The position values for the angles known through the artificial intelligence algorithm were taught to the system. Then, the position coordinates to be reached by this manipulator, which has four degrees of freedom, for the intermediate position coordinate values through artificial neural networks (ANN) have been obtained. It is expected that the device controlled by artificial intelligence will not be affected by the variables in parameter or force changes requiring high working performance. With the control of the positions through ANN, it has been ensured that the position control of the tapping robot manipulator is predicted based on artificial intelligence techniques depending on the angle values of the limbs, and the robot is prevented from going to a position that is on a different trajectory. Accordingly, the robot arm has been made controllable with ANN techniques. With ANN modelling, the position of the end point to perform the tapping process was estimated with high reliability. For future research, a rough simulation was made to see whether the end point would go to a different position in space.

Keywords: Artificial neural networks; Kinematic analysis; Tapping machine.

INTRODUCTION

Tapping is a process that is actively used in opening screws in holes (Bhowmick *et al.*, 2010). Having more than one cutter and threads in the desired forms around them are the characteristic features of taps. Cutters may be cylindrical or conical (Bhowmick *et al.*, 2010). According to the Turkish Standards Institution (TSE), a tap is defined as a cutter with screw threads on it that is used for cutting screws in holes that are drilled in materials such as metal and plastics via drilling bits or are brought to a specific point through turning (Avuncan, 1998; Uzun & Korkut, 2012).

Tapping, in which the screw is cut by combining the tool's rotational and advancing movements, is basically the performance of drilling operations. It can be performed by different methods. Some of these methods are drilling machines, machining centers, lathe centers equipped with rotary heads, conventional turning, and manual something (Akkurt, 1992). Today's developments in robotics allow the use of manipulators with automatic control in tapping.

In the industrial sector, the need for robot manipulators has increased as a result of the development of robotized tasks (Baizid *et al.*, 2016). These needs arise from the definition of the task to the implementation phase. In recent years, major developments have occurred in industrial automation. As a result, robotized tasks have been used and controlled more systematically and efficiently (Baizid *et al.*, 2016; Brog˚ardh, 2009). This has brought about different quests for manipulator control. Among the control methods developed in robotics in the course of time are show and teach, explicit robot programming languages, task-level programming languages, and offline programming methods. Computer programming predominates in the modern control approach (Craig, 2005).

Control methods such as P type control, PI type control, PID control, fuzzy logic, and artificial neural networks are the main methods of control technology today (Hoque *et al.*, 1995; Buja, 1993). These methods are accompanied by some disadvantages. Creating a mathematical model is a challenging field of engineering, and a mathematical modeling that is sure to be accurate is required to work with PI or PID control systems. Because of this, sometimes PI or PID controlled systems cannot meet the demands of high performance systems (Hoque *et al.*, 1995).

The characteristic of artificial neural networks is a nonlinear structure between inputs and outputs. This characteristic of artificial neural networks ensures that they are not affected by parameter or load changes in high performance systems (Hoque *et al.*, 1995). At the same time, with this structure, they can understand the approach between input and output with the training data and then yield the accurate output value when an input value different from such training data is given (Buja, 1993).

In their study, Pashkevich *et al.* examined artificial neural networks for a collision-free trajectory planning of a welding robot. The welding robot takes the irregular obstacles encountered by its units into consideration with the developed trajectory planning algorithm (Pashkevich *et al.*, 2006). Zhang et al. have studied optimal trajectory planning using genetic algorithms in a three-dimensional working space of a mobile robot. The positions and heights of the obstacles are known, and the trajectory is optimized with the help of genetic algorithm (Zhang *et al.*, 2008). Menasri *et al.* have proposed a new trajectory planning method in case of obstacles on the trajectory of a robot arm. In this study, the trajectory is disjointed, and the new position of the end effector is investigated in every step in Cartesian space to reach the final step. With the help of this technique, the position of the obstacles and the number of the constraints in the formulation of the problem are adapted (Menasri *et al.*, 2015). In Chaki *et al.*'s research, an integrated model of artificial neural networks (ANNs) and nondominated sorting genetic algorithm (NSGAII) for prediction and optimization of quality characteristics during pulsed Nd:YAG laser cutting of aluminum alloy (Chaki *et al.*, 2015). In Chen and Mill's research, an approach for improving the performance of industrial robots using multilayer feedforward neural networks is presented (Chen & Mills, 1997). In another study, integrated recurrent neural networks are applied as a feedforward controller for PUMA560 manipulator (Nagata & Watanabe, 2011). In another study, various experiments were carried out on human robot and industrial robot. These experiments validate the architecture (Bhat *et al*., 2016).

ANN method has been widely used in the scientific studies such as manufacturing processes, electronic, computer science, and similar other studies. ANN is an extremely useful way of modeling of experimental data modelling without conducting any experiments. These methods provide high reliability of predicting experimental test results (Aia *et al.,* 2016; Aia *et al*., 2016; Sahin, 2014; Ozkan & Toktas, 2016; Ozkan, 2016; Ozkan *et al.*, 2014; Ozkan, 2013; Ozkan, 2012).

Currently, manufacturers seek low-cost, fast, and productive solutions to meet the ever-growing demand. One of the principal concerns is reducing labor costs. The need for ensuring quality in products and standardization in such quality accompanies it. Meanwhile, saving time is a fundamental expectation. All these make automation more valuable. Thanks to automation systems, labor costs can be reduced; time can be saved; and standardization can be achieved in the work performed.

Tapping manipulator is a special industrial robot designed for tapping. Tapping is a necessity that arises in most manufacturing processes requiring assembly. Thanks to this manipulator, the tapping process has been transformed into a controllable form with computer support. Thus, it will allow for flexibility in system organization on integrated production lines.

In this study, a manipulator that can fulfill the function of tapping has been designed and manufactured. Kinematic analysis has been performed for this device, and the position analysis of the end effector has been performed. Accuracy analysis of the position equations has been performed. Artificial intelligence algorithm has been created, and the known position values have been taught to the system. In this way, the system has been equipped with the ability to make validations for unknown intermediate positions.

ROBOTAPP MECHANICAL DESIGN FEATURES

The tapping machine has three degrees of freedom up to the wrist. In addition to this design's wrist, a servo motor and a hydraulic motor have been added. Thus, two more degrees of freedom were obtained. Three servo motors positioned from the principal axis to the wrist have been used to position the wrist and make it follow the desired trajectory. With the servo motor in the wrist, the axis shift of the end effector has been blocked. Thanks to the hydraulic motor placed on the end effector, the tap head has been made rotational.

Figure 1 shows the mechanical structure of robotapp. As can be seen in the figure, robotapp manipulator has three bar mechanisms in its upper arm mechanical structure. With this mechanism, the end effector is always positioned perpendicular to the working plane. Thus, the end effector servo motor giving the fourth degree of freedom can be controlled separately.

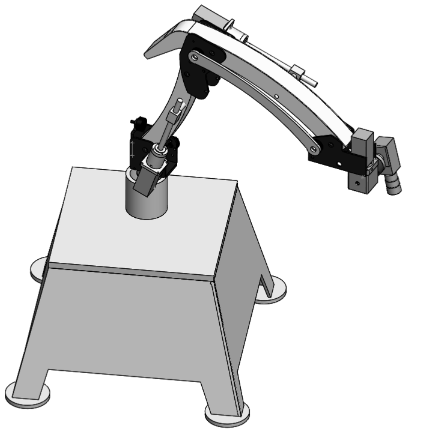


Figure 1. Tapping machine (ROBOTAPP).

In Figure 2, the joints of the mechanical structure and the angular positions and axial locations of these joints are seen. The end effector has been positioned through the first three joints (θ1, θ2, and θ3). With the help of the fourth joint (θ4), solutions for angular threading problems are provided.

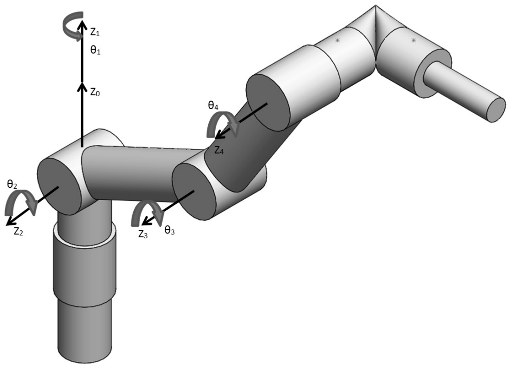


Figure 2. Positioning, rotation angles, and parametric measurements of the axes for the tapping machine.

ROBOTAPP KINEMATIC POSITION ANALYSIS

Denavit–Hartenberg principle has been used for kinematic analysis. There are separate vector representations for each axis, and the matrices are defined for each vector representation. Table 1 shows the table of the Denavit–Hartenberg parameters defined for the tapping machine (Malek & Othman, 1999). Transformation matrices have been formed based on this table. The fixed parameters of the chart have been used in the operations as l2=571.34 mm and l3=649.97 mm.

Table 1. The Denavit–Hartenberg parameters for the tapping machine (Malek & Othman, 1999).

Axis	α_{i-1}	a_{i-1}	d_i	θ_i
1	$\pi/2$	0	0	θ_1
2	0	l_2	0	θ_2
3	0	l_3	0	θ_3
4	0	0	0	θ_4

$$ {}^{0}_{3}T = \begin{bmatrix} r_{11} & r_{12} & r_{13} & P_x \\ r_{21} & r_{22} & r_{23} & P_y \\ r_{31} & r_{32} & r_{33} & P_z \\ 0 & 0 & 0 & 1 \end{bmatrix} = {}^{0}_{1}T\,{}^{1}_{2}T\,{}^{2}_{3}T \qquad (1)$$

When the transformation matrices 1, 2, and 3 are written in their places in the equation 1, matrix components 2, 3, and 4 emerge (Balkan, T., et. all, 2000). Here, Px, Py, and Pz represent the position of the end effector, while r11,r12,r13,….,r33 represent the rotation matrix components.

$$P_x = \cos(\theta_1) * (l_3 * \cos(\theta_2 + \theta_3) + l_2 * \cos(\theta_2)) \qquad (2)$$

$$P_y = \sin(\theta_1) * (l_3 * \cos(\theta_2 + \theta_3) + l_2 * \cos(\theta_2)) \qquad (3)$$

$$P_z = l_3 * \sin(\theta_2 + \theta_3) + l_2 * \sin(\theta_2) \qquad (4)$$

Position-dependent angle equations for the tapping machine have been obtained using the inverse kinematic method. Equation 5 presents the transformation matrix existing between the basic joint and the axis where the end effector is located. This matrix is shown here symbolically, but it actually represents numerical values. It is required to represent the variables in numerical values using Equation 5-7 (Chikhaoui, *et al.*, 2016).

$$ {}^{B}_{W}T = \begin{bmatrix} r_{11} & r_{12} & r_{13} & P_x \\ r_{21} & r_{22} & r_{23} & P_y \\ r_{31} & r_{32} & r_{33} & P_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad (5)$$

The matrix in Equation 5 can also be written in the format seen in equation 6 (Chikhaoui *et al.*, 2016).

$$ {}^{B}_{W}T = {}^{0}_{1}T\,{}^{1}_{2}T\,{}^{2}_{3}T \qquad (6)$$

$$ {}^{0}_{1}\mathrm{T}^{-1}\,{}^{B}_{W}\mathrm{T} = {}^{1}_{0}\mathrm{T}^{-1}\,{}^{0}_{1}\mathrm{T}\,{}^{1}_{2}\mathrm{T}\,{}^{2}_{3}\mathrm{T} \qquad (7)$$

Using equation 7, equations θ1, θ2, and θ3 have been obtained as follows using Equations 8–14 (Siciliano *et al.*, 2009):

$$\theta_1 = \pi + \mathrm{a}\tan 2(\mathrm{P_y}, \mathrm{P_x}) \qquad (8)$$

$$C_3 = \frac{P_x^2+P_y^2+P_z^2-l_3^2-l_2^2}{2l_2l_3} \qquad (9)$$

$$s_3 = \mp\sqrt{1-c_3^2} \qquad (10)$$

$$\theta_3 = atan2(s_3; c_3) \qquad (11)$$

$$s_2 = \frac{(l_2+l_3c_3)P_z - l_3s_3\sqrt{P_x^2+P_y^2}}{P_x^2+P_y^2+P_z^2} \qquad (12)$$

$$c_2 = \frac{(l_2+l_3c_3)\sqrt{P_x^2+P_y^2}+l_3s_3P_z}{P_x^2+P_y^2+P_z^2} \qquad (13)$$

$$\theta_2 = atan2(s_2; c_2) \qquad (14)$$

In the tapping machine, θ4 has been externally controlled. For this reason, θ4 has not been included in the calculations. Three-bar mechanism applied in the design of the tapping machine ensures that the end effector is perpendicular to the floor for each angle value of the lower limbs of the system. Therefore, input angles (θi) must be externally provided to the end effector for special cases, that is angular operations, with the help of the fourth servo motor.

Tapping machine is a mechanical system with an automation direction. The angular position equations are obtained through solution of equation 7 (in page 7,8) as equation 8,9,10,11,12,13,14. There are angular positions, where equation 7's solutions reach infinite values. These positions are called singular points and are determined by singular point analysis (Craig, 2005). There is a method to follow for avoiding these points. If the determinant of the Jacobian matrix is equal to zero, singularity emerges. The solution of this equation gives information to the designer about the points to be avoided (Hijazi, *et al.*, 2016) [29].

$$|J| = \begin{vmatrix} (-l_3s1c23 - l_2s1c2) & (-l_3c1s23 - l_2c1s2) & -l_3c1s23 \\ (l_3c1c23 + l_2c1c2) & (-l_3s1s23 - l_2s1s2) & -l_3s1s23 \\ 0 & (l_3c23 + l_2c2) & l_3c23 \\ (-l_3s1c23 - l_2s1c2) & (-l_3c1s23 - l_2c1s2) & -l_3c1s23 \\ (l_3c1c23 + l_2c1c2) & (-l_3s1s23 - l_2s1s2) & -l_3s1s23 \end{vmatrix} \qquad (15)$$

$$\det(J) = -l_2l_3s3(l_2c2 + l_3c23) \qquad (16)$$

The following results have been obtained by equalizing the determinant to zero in order to avoid the singular point:

Since l2 and l3 can never be zero, either of the following has to be true here: s3=0 or l_2 c2+l_3 c23=0. Considering these principles, θ3=0 and θ3=π must not occur. In addition, Px=Py=0 values must be avoided. The geometry of the manipulator does not allow this, either.

There are angle values that should not be reached technically. These values have completely been determined by the geometric properties of the manipulator. The working ranges of θ2 and θ3 based on geometric constraints are as follows:

$$\theta2=0 \sim 89^{o} \text{ ve } \theta3= -90^{o} \sim -160^{o} \qquad (17)$$

MODELLING THE POSITION OF THE ROBOT MANIPULATOR WITH ANN

Neural networks consist of neurons. Their shape and size may vary depending on the function they are to fulfill. Understanding the way, a neuron works helps to build ANN. ANN can be seen as a black box containing sets of hierarchical neurons that produce output in return for certain input values. Each processor in ANN includes collecting and processing the data and sending the results to the related component. The whole process within the ANN structure can be examined in terms of inputs, weights, sum function, and activation function. According to Figure 3, the things to consider are as follows:

Inputs (I)

Weights (w)

Sum function (Σ)

Transfer (activation) function (f)

Outputs (O) (Ozkan & Toktas, 2016; Ozkan, 2016; Ozkan *et al.*, 2014; Ozkan, 2013; Ozkan, 2012).

With the data modeled in the study, four-axis tapping robot has been modelled on Solidworks software first. These data have confirmed the position values of each limb with engineering approach. With this model, the positions of all the linkages have been collected in an excel file as a data set. A code has been prepared on Matlab software, and the data obtained have been subjected to a training process. In this study, tapping robot motions parameters were defined. This robot contains tapping tool on it. Robot end linkage positions (x, y and z) were predicted without using any engineering formulae. In this regard, a dataset composed of 209 rows and 11 columns has been prepared. Approximately 30% of this dataset has been used for testing and 70% for network training. In this ANN model contains inputs and outputs. Inputs were robot linkage positions (x, y and z) and each angles of robot linkage (θi) and outputs were end members' location (xn, yn and zn) ANN model was built with architecture of LM (Levenberg–Marquardt) algorithm and MLP (MultiLayer Perception).

Many different model iterations were tried to determine the best ANN model (Table 2). For this aim MLP and RBF model were iterated. In this stage BFGS, RBFT and LM learning algorithms were also modelled in MATLAB ANN Tool. MATLAB Tool is given the model performance as statistically. A code has been prepared in MATLAB ANN Tool. This code has ANN model and also statistical calculations for each ANN iteration. Model results were also calculated in a excel file by statistically. The best model was determined as MLP and LM training algorithm. The best model has been defined 4- hidden-layer (Tansig (12) – Tansig (11) – Logsig (135) – Purelin (3)) MLP and LM training algorithm. This code is shown Training, Test and Validation reliability for all data (Table 2). The statistical methods of RMSE, R2, and MEP % values have been obtained using Eq. 18-20 (Ozkan, 2016; Ozkan *et al.*, 2014; Ozkan, 2013; Ozkan, 2012).

$$RMSE = \left((1/p) \sum_j \left| t_j - o_j \right|^2 \right)^{1/2} \tag{18}$$

$$R^2 = 1 - \left(\frac{\sum_j (t_j - o_j)^2}{\sum_j (o_j)^2} \right) \tag{19}$$

$$MEP = \frac{\sum_j \left(\frac{t_J - o_j}{t_j} \times 100 \right)}{p} \tag{20}$$

where t is the target value, o the output, and p the number of samples.

Table 2. ANN model iterations.

Number of Iteration	ANN Model	Training Performance	Test Performance	Validation Performance	Training Algorithm	Error Function	Hidden Activation	Output Activation
1	MLP 6-35-3	0.997102	0.992434	0.995052	BFGS 144	SOS	Tanh	Logistic
2	MLP 6-45-3	0.990159	0.975714	0.985681	BFGS 194	SOS	Exponential	Exponential
3	MLP 6-122-3	0.997523	0.992026	0.995407	BFGS 207	SOS	Logistic	Exponential
4	RBF 6-4-3	0.495781	0.366679	0.627060	RBFT	SOS	Gaussian	Identity
5	RBF 6-4-3	0.694464	0.715604	0.651308	RBFT	SOS	Gaussian	Identity
6	RBF 6-4-3	0.710787	0.639707	0.724063	RBFT	SOS	Gaussian	Identity
7	RBF 6-4-3	0.632107	0.583729	0.628993	RBFT	SOS	Gaussian	Identity
8	RBF 6-4-3	0.640763	0.608003	0.664539	RBFT	SOS	Gaussian	Identity
9	RBF 6-4-3	0.663639	0.667086	0.749902	RBFT	SOS	Gaussian	Identity
10	MLP 6-29-3	0.993101	0.984830	0.991364	BFGS 117	SOS	Exponential	Logistic
11	MLP 6-4-3	0.915133	0.895702	0.942572	BFGS 71	SOS	Tanh	Tanh
12	RBF 6-12-3	0.831743	0.845800	0.781032	RBFT	SOS	Gaussian	Identity
13	MLP 12-11-135-3	0.999840	0.996850	0.999270	LM	SOS	Exponential	Logistic

In the ANN model, tansig, logsig, and purelin have been used as transfer function (f) (Eq. 21-24). Their equations are given below (Figure 3) (Ozkan, 2016; Ozkan *et al.*, 2014; Ozkan, 2013; Ozkan, 2012).

$$\mathrm{NETi} = \Sigma \mathrm{wij} \cdot \mathrm{xj} + \mathrm{wbi} \qquad (21)$$

$$a = \mathrm{tansig}\ (\mathrm{n}) = \frac{2}{(1+e^{-2n})} - 1 \qquad (22)$$

$$a = \mathrm{logsig}\ (\mathrm{n}) = \frac{2}{(1+e^{-n})} \qquad (23)$$

$$a = purelin\ (n) \qquad (24)$$

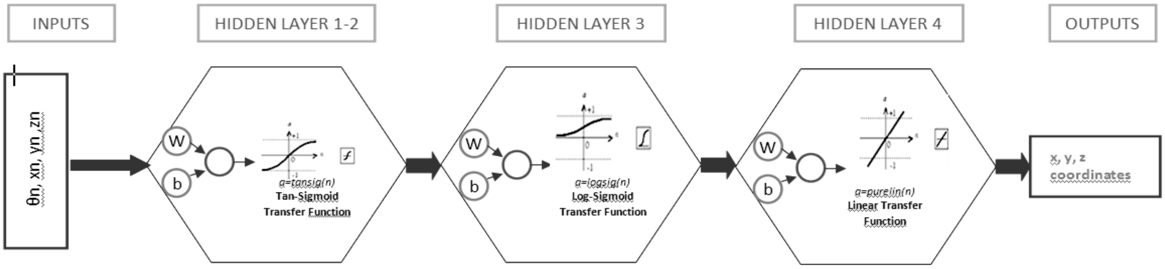


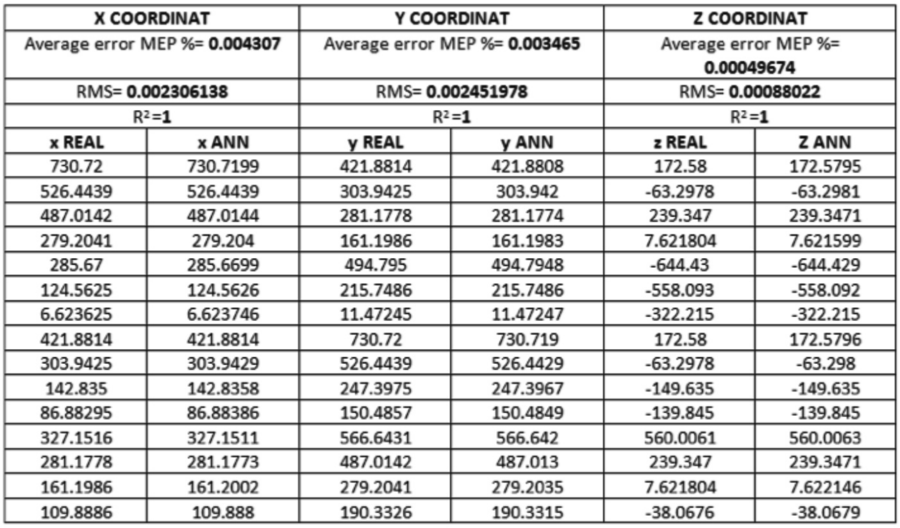
Figure 3. Basic ANN Model Structure (Ozkan, 2016; Ozkan, 2012).

Table 3. The position values obtained based on the angle values, and obtained through the mathematical model acquired from kinematic equations.

θ_1 (degree)	θ_2 (degree)	θ_3 (degree)	x COORDINAT (mm)	y COORDINAT (mm)	z COORDINAT (mm)
30	60	-90	730.72	421.8814	172.58
30	60	-120	526.4439	303.9425	-63.2978
30	89	-120	487.0142	281.1778	239.347
30	89	-150	279.2041	161.1986	7.621804
60	0	-90	285.67	494.795	-644.43
60	0	-120	124.5625	215.7486	-558.093
60	0	-150	6.623625	11.47245	-322.215
60	60	-90	421.8814	730.72	172.58
60	60	-120	303.9425	526.4439	-63.2978
60	60	-150	142.835	247.3975	-149.635
60	60	-160	86.88295	150.4857	-139.845
60	89	-90	327.1516	566.6431	560.0061
60	89	-120	281.1778	487.0142	239.347
60	89	-150	161.1986	279.2041	7.621804
60	89	-160	109.8886	190.3326	-38.0676

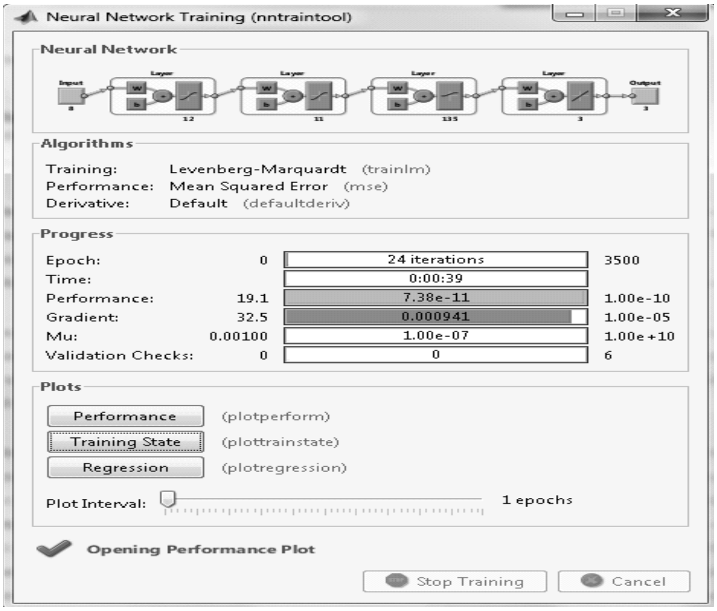
The results of the comparison through regression analysis of the validated values obtained through ANN and the random data, some of which are randomly chosen and showed in Table 3 and 4, that have been obtained from engineering equations and in which the positions of the robot in Solidworks software have been tested for the positions x, y, and z are as follows: R2=1 for training; R2=0.99685 for test, and R2=0.99927 for validation (Table 4).

Table 4. Comparison and Regression Analysis of Experiment-ANN Data (Test).



X COORDINAT		Y COORDINAT		Z COORDINAT	
Average error MEP %= **0.004307**		Average error MEP %= **0.003465**		Average error MEP %= **0.00049674**	
RMS= **0.002306138**		RMS= **0.002451978**		RMS= **0.00088022**	
R^2=**1**		R^2=**1**		R^2=**1**	
x REAL	**x ANN**	**y REAL**	**y ANN**	**z REAL**	**Z ANN**
730.72	730.7199	421.8814	421.8808	172.58	172.5795
526.4439	526.4439	303.9425	303.942	-63.2978	-63.2981
487.0142	487.0144	281.1778	281.1774	239.347	239.3471
279.2041	279.204	161.1986	161.1983	7.621804	7.621599
285.67	285.6699	494.795	494.7948	-644.43	-644.429
124.5625	124.5626	215.7486	215.7486	-558.093	-558.092
6.623625	6.623746	11.47245	11.47247	-322.215	-322.215
421.8814	421.8814	730.72	730.719	172.58	172.5796
303.9425	303.9429	526.4439	526.4429	-63.2978	-63.298
142.835	142.8358	247.3975	247.3967	-149.635	-149.635
86.88295	86.88386	150.4857	150.4849	-139.845	-139.845
327.1516	327.1511	566.6431	566.642	560.0061	560.0063
281.1778	281.1773	487.0142	487.013	239.347	239.3471
161.1986	161.2002	279.2041	279.2035	7.621804	7.622146
109.8886	109.888	190.3326	190.3315	-38.0676	-38.0679

In the ANN model, three output values (xo, yo and zo) in return for eight inputs (θi, xi, yi, zi), the coordinate values in the x, y and z axes have been taught at the same time, and the ANN results have been compared with the experimental data. In the ANN model, tansig, logsig, and purelin functions (8+ 12 + 11 + 135 + 3) have been used respectively. In this model: 8 demonstrates number of inputs, 12 (tansig)+11(tansig)+135(logsig) and 3 (purelin) and number of outputs. The network structure of the ANN model obtained through the Matlab program is shown in Figure 4. The best training performance is displayed in Figure 5. Regression analysis is presented in Figure 6.



Algorithms

Training: Levenberg-Marquardt (trainlm)
Performance: Mean Squared Error (mse)
Derivative: Default (defaultderiv)

Progress

Epoch:	0	24 iterations	3500
Time:		0:00:39	
Performance:	19.1	7.38e-11	1.00e-10
Gradient:	32.5	0.000941	1.00e-05
Mu:	0.00100	1.00e-07	1.00e+10
Validation Checks:	0	0	6

Plots
Performance (plotperform)
Training State (plottrainstate)
Regression (plotregression)
Plot Interval: 1 epochs
Opening Performance Plot
Stop Training
Cancel

Figure 4. The network structure of the ANN model.

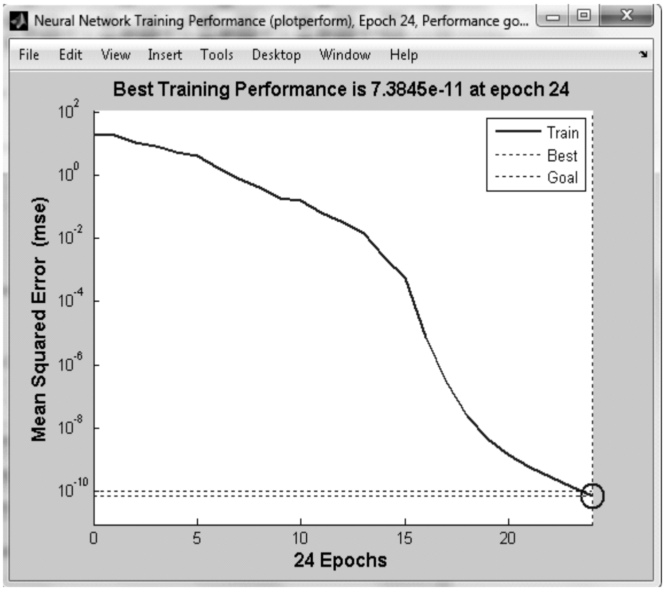


Figure 5. The best training performance.

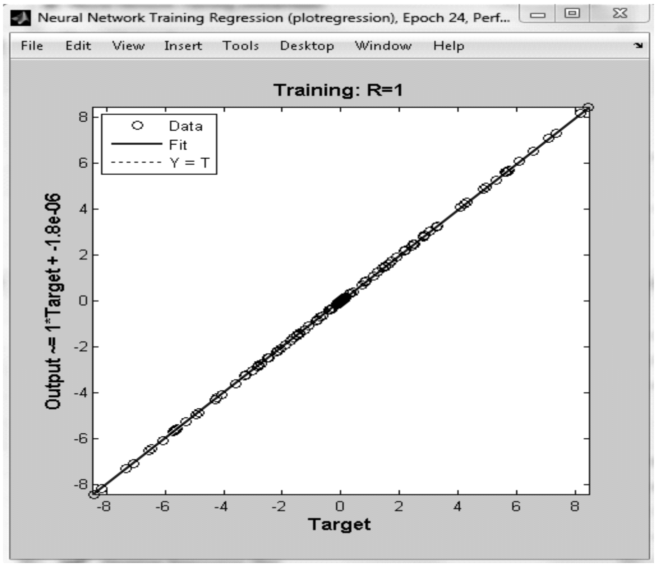


Figure 6. Regression analysis.

The graphs given in Figures 6 and 7 show the consistency of the output validations obtained as a result of the training of the input data through ANN. The regression analysis in Figure 6 indicates that the validation results have been approximated by 1. This value shows that the ANN program yields nearly one hundred percent accurate results. The "Training R: 1" value given here indicates to what extent the experimental data correspond to the data obtained from ANN. Figure 7 presents best validation, training, and test comparison. According to the figure, as the amount of error in the training value decreases, the validation and test abilities of ANN increase, which is consistent with the literature. Figure 8 presents the comparison of ANN model's training, validation, test, and overall results. The regression analysis values of the training, test, and validation values are seen. The prepared software is seen to have a validation power of 0.99927 and a test power of 0.99685. The prepared software has an ability to validate the robot position at values extremely close to the reality. Figure 9 shows error histogram with 20 bins. The gathering of residual values in a specific area means that training, test, and validation values are extremely close to the reality. As these residual values are small values, it can be regarded as a proof that no problem will be caused in the positioning of the robot arm.

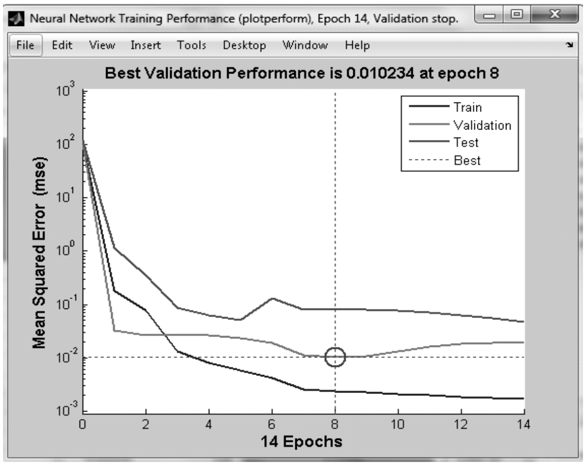


Figure 7. Training, validation, and test comparison.

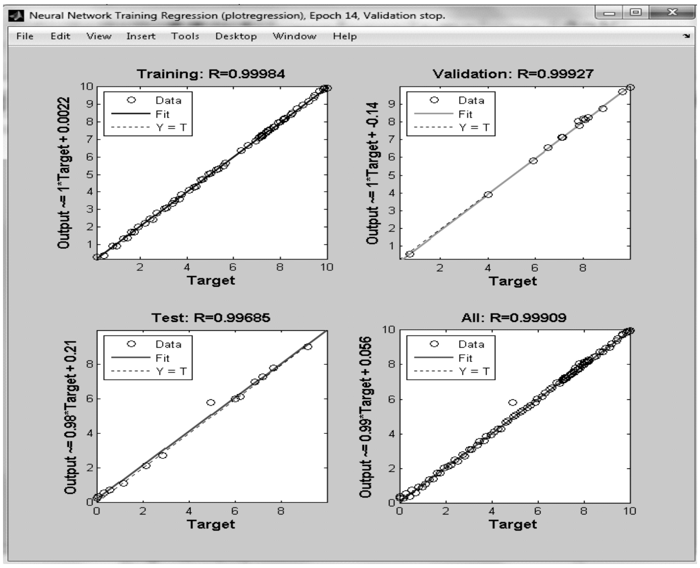


Figure 8. The comparison of ANN model’s training, validation, test, and overall results.

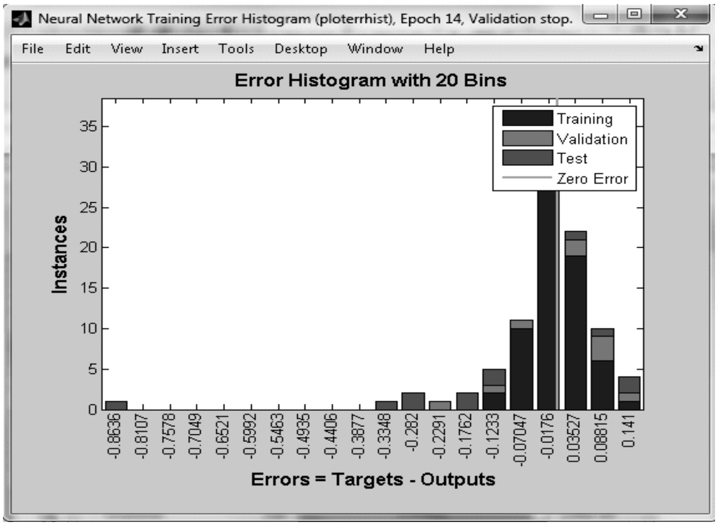


Figure 9. Error Histogram with 20 Bins.

CONCLUSION

Threaded connections are among the primary methods specifically used for industrial assembling operations. As can be understood from previous research carried out, tapping constitutes 22% of machining operations. Investigation of the systems that can perform tapping in a short period of time is important to conduct the process of opening threads in holes with a mechanical structure of ideal quality by saving time. Smart work assistants allow this in today’s technology. In the present study, a special mechanical design has been created for the tapping operation, and a smart work assistant prototype has been manufactured out of this mechanical structure. Tapping is a sensitive process, and it is required even in working conditions with variable parameters. In such cases, what is offered by ready-made control cards or the complicated mathematical models needed in PI and PID controllers may fail to yield accurate results. In the current technology, ANN method allows sensitive processes to be carried out in varying conditions with more accurate results. Speed and reliability are highly important in assembly lines involving serial processes. The data obtained in the study have showed that ANN is an appropriate solution for the position control of the end effector. With ANN, the tapping machine has been taught the positions of the system for certain values. This has been a work complying with the logic of teaching with examples, which is the basic principle of ANN.

In the working space, the known position coordinates of the end effector have been taught to the system, and the unknown coordinates have been validated. Regression analysis and comparison of ANN data are given in Table 2. The consistency of the outputs obtained is remarkable. The regression analysis has indicated that the ANN program has yielded almost one hundred percent accurate results. Determining robot arm positions with mathematical equations is a Labor-intense and time-consuming task. It needs to the examination of tapping robot members real positions (x, y and z) in space. For this aim, it was compared with mathematical equations results and ANN results. A motion simulation has been performed with a comparison of the values obtained from the ANN, and the position the robot is to head for, thereby preventing the robot from performing tapping in a wrong position. In addition to their safe use of in many other areas, ANN are also seen to provide a lot of advantages in identifying positions in that they determine the positions of robot arms in advance, perform simulation, and are cheap and less laborious.

The position of the end point was modelled with high precision depending on the linkage angels made by the tapping machine during the process and the sizes of its parts, without any need for engineering equations. With the automation-supported feature of the tapping machine, the program to be prepared by the bench programmer can function as a control mechanism. The use of the ANN method prevents the bench operator from writing an incorrect code. In this way, time has been saved, and occupational accidents and sharp object-workpiece damages have been reduced. The use of robot in the tapping process has brought standardization in manufacturing regardless of the operator's professional knowledge and skills. The device has gained an ability to work in conditions that can be considered dangerous for people.

ACKNOWLEDGMENT

This study was supported by Republic of Turkey Ministry of Science, Industry and Technology, General Directorate of Science and Technology 0353.STZ.2013-2 (SAN-TEZ, Ankara, Turkey).

COMPLIANCE WITH ETHICAL STANDARDS

Funding: This study was supported by Republic of Turkey Ministry of Science, Industry and Technology, General Directorate of Science and Technology, (Research project No: 0353.STZ.2013-2).

Conflict of Interest: The authors declare that they have no conflict of interest.

REFERENCES

Aia, Y., Jianga, P., Shao, X., Wang, C., Li, P., Mi, G., Liu, Y. & Liu, W. 2016. An optimization method for defects reduction in fiber laser keyhole welding. Applied Physics A. **122**(1): 31.

Aia, Y., Jianga, P., Shaoa, X., Wangb, C., Lia, P., Mib, G., Liua, Y. & Liua, W. 2016. A defect-responsive optimization method for the fiber laser butt welding of dissimilar materials. Materials & Design, **90**: 669-681.

Akkurt, M. 1992. Machining methods and machine tools (Talaş Kaldırma Yöntemleri ve Takım Tezgahları). Turkey: Birsen Yayınevi.

Avuncan, G. 1998. Economy of Cutting and Cutting Tools (Talaş Kaldırma Ekonomisi ve Kesici Takımlar). Gebze (Turkey), Mavi Tanıtım ve Pazarlama Ltd. Şti.

Baizid, K., Ćuković, S., Iqbal, J., Yousnadj, A., Chellali, R., Meddahi, A., Devedžić, G. & Ghionea, I. 2016. IRoSim: Industrial Robotics Simulation Design Planning and Optimization platform based on CAD and knowledgeware Technologies. Robotics and Computer-Integrated Manufacturing, **42**: 121-134.

Balkan, T., Özgören, M.K., Arıkan, M.S. & Baykurt, H.M. 2000. A method of inverse kinematics solution including singular and multiple configurations for a class of robotic manipulators. Mechanism and Machine Theory, **35**(9): 1221-1237.

Bhat, A.A., Akkaladevi, S.C., Mohan, V., Eitzinger, C. & Morasso, P. 2016. Towards a learnt neural body schema for dexterous coordination of action in humanoid and industrial robots. Autonomous Robots, **41**(4): 945-966.

Bhowmick, S., Lukitsch, M.J. & Alpas, A.T. 2010. Tapping of Al-Si alloys with diamond-like carbon coated tools and minimum quantity lubrication. Journal of Materials Processing Technology, **210**(15): 2142-2153.

Brog°ardh, T. 2009. Robot Control Overview: An Industrial Perspective. Modeling, Identification and Control, **30**(3): 167-180.

Buja, G.S. 1993. Neural Network Implementation of a Fuzzy Logic Controller. Industrial Electronics, Control, and Instrumentation, 1993. Proceedings of the IECON '93., International Conference on: 414-417.

Chaki, S., Bathe, R.N., Ghosal, S. & Padmanabham, G. 2015. Multi-objective optimisation of pulsed Nd: YAG laser cutting process using integrated ANN-NSGAII model. Journal of Intelligent Manufacturing, 1-16.

Chen, P.C. & Mills, J.K. 1997. Synthesis of Neural Networks and PID control for performance improvement of industrial robots. Journal of Intelligent and Robotic Systems, **20**(2): 157-180.

Chikhaoui, M.T., Rabenorosoa, K. & Andreff, N. 2016. Kinematics and performance analysis of a novel concentric tube robotic structure with embedded soft micro-actuation. Mechanism and Machine Theory, **104**: 234-254.

Craig, J.J. 2005. Introduction Robotics Mechanics and Control. New Jersey: Prentice.

Hijazi, A., Brethé, J.F. & Lefebvre, D. 2016. Singularity analysis of a planar robotic manipulator: Application to an XY-Theta platform. Mechanism and Machine Theory, **100**: 104-119.

Hoque, M.A., Zaman, M.R. & Rahman, M.A. 1995. Artificial neural network based permanent magnet DC motor drives. Industry Applications Conference, 1995. Thirtieth IAS Annual Meeting, IAS '95., Conference Record of the 1995 IEEE: 98-103.

Malek, K.A. & Othman, S. 1999. Multiple sweeping using the Denavit–Hartenberg representation method. Computer-Aided Design, **31**(9): 567-583.

Menasri, R., Nakib, A., Daachi, B., Oulhadj, H. & Siarry, P. 2015. A trajectory planning of redundant manipulators based on bilevel optimization. Applied Mathematics and Computation, **250**: 934-947.

Nagata, F. & Watanabe, K. 2011. Adaptive Learning with Large Variability of Teaching Signals for Neural Networks and Its Application to Motion Control of an Industrial Robot. International Journal of Automation and Computing, **8**(1): 54-61.

Ozkan, M.T. 2012. Notch sensitivity factor calculation in the design of shafts using artificial neural network system. Energy Education Science and Technology Part A: Energy Science and Research, **30**(1): 621-630.

Ozkan, M.T. 2013. Experimental and artificial neural network study of heat formation values of drilling and boring operations on Al 7075 T6 workpiece. Indian J Eng Mater Sci. **20**: 259-268.

Ozkan, M.T. 2016. Surface roughness during the turning process of a 50CrV4 (SAE 6150) steel and ANN based modeling. Materials Testing, **57**(10): 889-896.

Ozkan, M.T., Cengiz, E. & Ismail, S. 2014. Determination of the Notch Factor for Shafts Under Torsional Stress with Artificial Neural Networks. Materiali in tehnologije, **48**(1): 81-90.

Ozkan, M.T. & Toktas, I. 2016. Determination of the stress concentration factor (Kt) in a rectangular plate with a hole under tensile stress using different methods. Materials Testing, **58**(10): 839-847.

Pashkevich, A., Kazheunikau, M. & Ruano, A.E. 2006. Neural network approach to collision free pathplanning for robotic manipulators. International Journal of Systems Science, **37**(8): 555-564.

Sahin, I. 2014. Prediction of Surface Roughness of Al/SiC Composite Material with Artificial Neural Networks, Journal of the Faculty of Engineering and Architecture of Gazi University, **29**(1): 209-216.

Siciliano, B., Sciavicco, L., Villani, L. & Oriolo, G. 2009. Robotics: Modelling, Planning and Control. London: Springer.

Uzun, G. & Korkut, İ. 2012. Effect of cutting method and cutting parameters on threading with tapping (Kılavuz ile Vida Açma İşlemine Kesme Yönteminin ve Kesme Parametrelerinin Etkileri). 3. UTIS, 04-05. October 2012, Ankara, Turkey, 264-273, Resource Document. 3. UTIS file:///C:/Users/PC-01/Downloads/264-273%20(2).pdf. Accessed 20 February 2017.

Zhang, H., Liu, M., Liu, R. & Hu, T. 2008. Path planning of robot in three-dimensional grid environment based on genetic algorithms. 2008 7th World Congress on Intelligent Control and Automation, 1010-1014.