A NEURAL NETWORK APPROACH FOR PREDICTING KINEMATIC ERRORS SOLUTIONS FOR TROCHOIDAL MACHINING

Farid Bettine[1], Hacene Ameddah[2], Rabah Manaa[3]

[1,3]Laboratory of Research in Productivity (LRP), Faculty of Technology, Department of Mechanical, University of Batna 2, Batna 0500, Algeria
[2]Laboratory of Innovation in Construction, Eco-design, and Seismic Engineering (LICEGS), Faculty of Technology, Department of Mechanical,University of Batna 2, Batna 0500, Algeria

Corresponding author: Hacene Ameddah, hacamed@gmail.com

Abstract: The prediction of machining accuracy of a five-axis machine tool is a vital process in precision manufacturing for machining a hard and free form surfaces. This work presents a novel approach for predicting kinematic errors solutions in five-axis machine for trochoidal milling strategy. This approach is based on Artificial Neural Network (ANN) for trochoidal milling machining strategy. We proposed a multi-layer perceptron (MLP) model to find the inverse kinematics solution for a five-axis machine. The data sets for the neural-network model are obtained using kinematics software. The solution of each neural network is estimated using inverse kinematics equation of the machine tool to select the best one. As a result, the Neural Network implementation improves the performance of the learning process. For this, numerical study of trochoidal strategy and experimental results are presented with aims to full milling and to ensure a control of radial engagement. The experimental result shows the efficiency of the method by obtainning the toolpath and the machining possebility of this new type of strategy emerging.
Key words: Five-axis machine, kinematic errors, neural network, trochoidal strategy of machining.

1. INTRODUCTION

Five-axis machine tools have more error sources compared with the 3-axis machine tools. Consequently, it is more difficult to determine the error sources of a five-axis machine tool due to the complexity of configuration [1]. In general, the source errors of five-axis machine tools can be classified into four types: geometric errors, kinematic errors, and thermally induced errors, deflection errors caused by cutting forces, and other errors, such as those cause d by servomotors, errors of machine axes rotation, or numerical control interpolation algorithmic errors [2].
This kinematics problem is known to be a nonlinear problem having multiple solutions; its complexity widely depends on the order of appearance and the relative orientation of the different joint (link) axis along the kinematic chain, and can be treated like those of robotic manipulators [3-4]. Many researchers have studied kinematics problem for different types of 5-axis machines. Farouki et al. [5] study the problem of determining the inputs to the rotary axes of a 5-axis CNC machine, such that relative variations of orientation between the tool axis and surface normal are minimized subject to the constraint of maintaining a constant cutting speed with a ball-end tool.
Moonet and al. [6] developed a methodology to evaluate and compensate for the kinematic error at the tip of the tool in multi-axis machine tools using screw theory. Hai-Yin Xu and al. [7] proposed kinematic chain used in the management of tool wear in five-axis machining. Sorby [8] used a homogeneous matrix in inverse kinematics to study the positioning of the machine axes close to the singular configurations of a 5-axis milling machine. Lamikiz and al. [9] used a homogeneous matrix to evaluate the consequences in the tool tip position in geometrical errors.
However, tracking errors of both rotary and translational axes are kinematically transmitted to the tool tip in five-axis machine tools, resulting in contouring errors between the commanded and the actual tool paths [10]. In this study, we identify the relation between workpiece coordinate and machine coordinate with the aims to applied trochoidal toolpath milling.
There are two types of kinematics: the first one is the forward kinematic problem (FKP) which is the mapping from joint space to Cartesian space (operation space); the FKP is to find the position and the orientation of the end-effector. Traditionally, there were three methods used to solve forward kinematics problem of robotic manipulators, which are geometric, algebraic and iterative methods [11]. This problem admits a single solution, which can be determined by simple matrix and vector

multiplications. The second one is the Inverse Kinematic Problem (IKP) is the mapping from Cartesian space to joint position; the IKP is to find the joint space from the pose of the tool.

The solution of IKP may give a set of solutions or in some cases no solution. That is why it is significant to apply an artificial neural network models.

Here artificial neural network (ANN) approach has been proposed to control the kinematic of five-axis machine tool. Artificial neural network method is used to learn the forward and the inverse kinematics equations of 5-axes CNC machining System. This method learns the functional relationship between input (Cartesian space) and output (joint space) based on a localized adaptation of the mapping [12,13]. The simulation and computation of inverse kinematics using multiple layer perceptrons (MLP) is particularly useful where less computation times are needed, such as in real-time adaptive machine tool control [14,15].

In this work we propose a procedure to resolve inverse kinematics of five-axis machine (TTTRR). Three linear axes (X, Y, and Z) and two rotary axes (A, C) comprise this machining, the table moves linearly in the X and Y directions while the head moves linearly in Z direction. The table also tilts about the X-axis and rotates about the Z-axis (angles A and C, respectively), using a structured multilayer perceptron (MLP), that can be trained quickly. The result of each network is assessed using inverse kinematics equations to extract information about inverse kinematic error. Otherwise stated, the tool orientation and position obtained for each link is used to compute the Cartesian coordinate for the end effector.

2. KINEMATIC MODEL OF A FIVE-AXIS MACHINE

To adequately control the position and orientation of the workpiece and the cutter, a kinematic model is required to describe the motion of the mechanism in mathematic. The mechanism of a five-axis machine can be seen as a chain of rigid bodies (or links) connected by the joints, which may be either revolute or prismatic.

The kinematics model of a five -axis machine can be seen as a combination of the individual kinematic models of the links. In the section, the forward kinematics model of the milling centre is found using the D-H method.

Figure 1 shows the kinematic chain. The end of the chain is attached to cutter tool (t), while the beginning of the chain is attached to work-piece (p). In addition to the joint t and p, there are five intermediate joints, which indicate the rotational, or the translational joints.

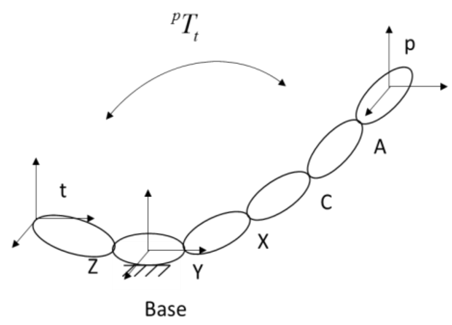


Fig. 1. The kinematic chain of the five-axis

Figure 2 shows the geometry of five-axis milling at its initial position. At this configuration, the origin of tool holder frame G is coincident the workpiece frame P on top of centre of the table.

The table of the machine has two revolute axes C and A, and two prismatic axes X and Y. The column of machine is fixed which has coordinate frame B and tool holder has a vertical prismatic axis Z.

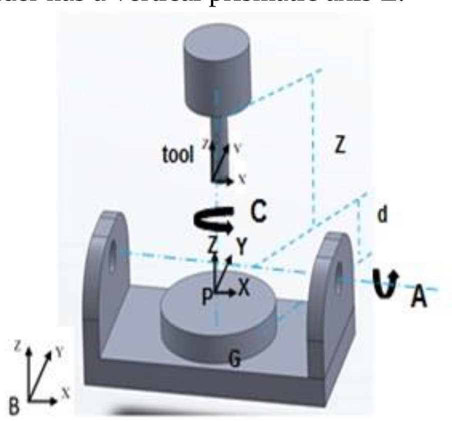


Fig. 2. Model the table of five-axis machine

The Denavit-Hartenberg (D-H) first introduced the spatial transformation between two successive links using a 4 × 4 D-H matrix.

$$^{i-1}T_i=Rot(Z,\theta_i)Trans(Z,b_i)Trans(X,a_i)Rot(X,\alpha_i) \quad (1)$$

Once the corresponding D-H parameters are defined, these parameters can by create directly from workpiece to tool according to (table 1).

Table 1. The D-H parameter Five-axis

Links	θ_i	a_i	b_i	α_i
$^pT_1$	$\pi/2$	0	0	π
$^1T_2$	C	0	d	π
$^2T_3$	A	0	0	0
$^3T_4$	$-\pi/2$	0	X	$\pi/2$
$^4T_5$	$\pi/2$	0	Y	$\pi/2$
^5T_t	π	0	Z+d	0

The D-H matrices for these links can be multiplied to each other to form a single transformation from p to t, with this transformation matrix can be related as $^pT_1$ to ^5T_t , is calculated from equation (1), so ^pT_t can be calculated as follow:

$$ {}^{P}T_t = {}^{P}T_1 \, {}^{1}T_2 \, {}^{2}T_3 \, {}^{3}T_4 \, {}^{4}T_5 \, {}^{5}T_t \quad (2) $$

Forward kinematics are used to calculate the coordinate system (P–system) by the position vectors $P = \left[P_x, P_y, P_z\right]^T$ and orientation $O = \left[O_i, O_j, O_k\right]^T$, with respect to machine axis variables $\theta_i^\circ, a_i, b_i, \alpha_i^\circ$. The component position and orientation relative to the machine base is given by the matrix D-H:

$$ {}^{p}T_t = \begin{bmatrix} \cos(C) & \cos(A)\sin(C) & \sin(A)\sin(C) & -X\cos(C) - Y\cos(A)\sin(C) + (Z+d)\sin(A)\sin(C) \\ -\sin(A) & \cos(A)\cos(C) & \sin(A)\cos(C) & X\sin(C) - Y\cos(A)\cos(C) + (Z+d)\sin(A)\cos(A) \\ 0 & -\sin(A) & \cos(A) & Y\sin(A) + (Z+d)\cos(A) - d \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3) $$

The position and rotation of the vectors of cutter tools relative to work-pieces is determined by the matrix D-H.

The first three rows of the last two columns of matrix (3) provide the tool orientation $O = \left[O_i, O_j, O_k\right]^T$ and position $P = \left[P_x, P_y, P_z\right]^T$.

The forward kinematics of the machines tool is:

$$ \begin{cases} O_i = \sin(A)\sin(C) \\ O_j = \sin(A)\cos(C) \\ O_k = \cos(A) \\ P_x = -X\sin(C) - Y\cos(A)\sin(C) + (Z+d)\sin(A)\cos(C) \\ P_y = X\sin(c) - Y\cos(A)\cos(C) + (Z+d)\sin(A)\cos(C) \\ P_z = Y\sin(A) + (Z+d)\cos(A) \end{cases} \quad (4) $$

Using the forward kinematics transformation from in equation (4), the reference tool path variables $\left(P_x, P_y, P_z, O_i, O_j, O_k\right)$ the coordinates system (P–system) is transformed into coordinates in the machine tool coordinate system (M-system) as follows:

$$ \begin{cases} A = \pm\arccos(K) \\ C = \arctan(O_i / O_j) \\ X = -P_x\cos(C) + P_y\sin(C) \\ Y = -P_x\cos(A)\sin(C) - P_y\cos(A)\cos(C) + (P_z + d)\sin(A) \\ Z = P_x\sin(A)\sin(C) + P_y\sin(A)\cos(C) + (P_z + d)\cos(A) - d \end{cases} \quad (5) $$

Using the Inverse kinematic equations from (5), tool coordinate system (M–system) obtained axis movements for poled kinematic inverse as shown in Figure 3.

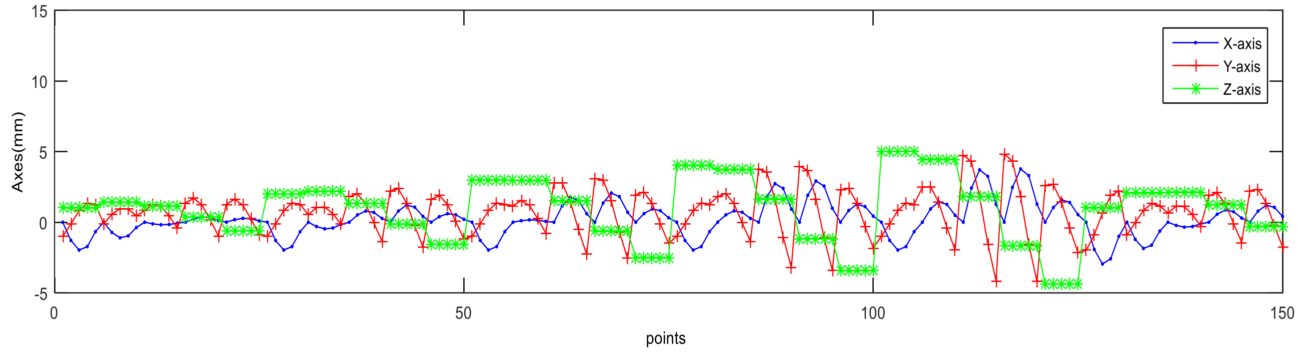


Fig. 3. Inverse kinematics results of (X, Y, Z) axis

3. TROCHOIDAL STRATEGY

Trochoidal machining is a type of machining trajectory emerging due to the increase in the performance of manufacturing means [17].

The principle of a trochoidal curve is to have the cutting tool describe a curve of continuous curvature, thus avoid to work in full matter. As we will show later, they allow controlling the axial engagement of the tool which leads to a better management of cutting forces.

From a mathematical point of view, the term trochoid refers to the curve obtained by combining a uniform circular motion and a uniform linear motion. For example, in Figure 4, the point B rotates uniformly around the point A, itself being animated by a linear motion.

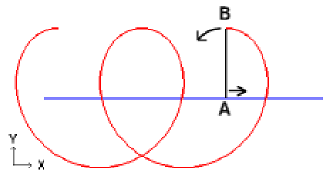


Fig. 4. Trochoidal movement

A parametric equation of a trochoid is given in Cartesian coordinates by the system of equations (6), using the following parameters: V, velocity of the center of the circle in its uniform rectilinear motion, R, radius of the circle (R> 0), ω pulsation (ω> 0) and t, parameter of the curve (t> 0).

$$ \begin{cases} x = V.t + R.\cos(w.t) \\ y = R.\sin(w.t) \end{cases} \quad (6) $$

Pocket dump is a central issue in rough milling. Trochoidal machining is a new type of strategy that

has found applications in rough machining of hard materials.
A trochoidal trajectory generation formulation has been developed. The results of this study lead to a better knowledge of this new machining strategy, with a view to its application to the roughing of light alloys.

Table 2. Characteristics mill and groove

Cutter diameter (mm)	20
Width of the groove (mm)	28
Length of the groove (mm)	120
Step (mm)	1.5

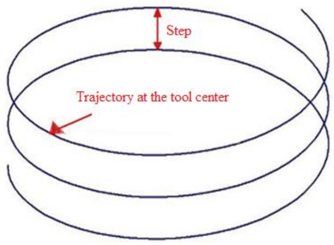


Fig. 5. Definition of the tool pitch

3.1 Results

The main advantage of the trochoidal path (Figure 6) is to present a continuous path radius leading the machining process to take place under favorable conditions (no impact, less marking of the part, ...).
For that a graph representing some points is represented (Figure 6) allowing to control the curvature of the segments.

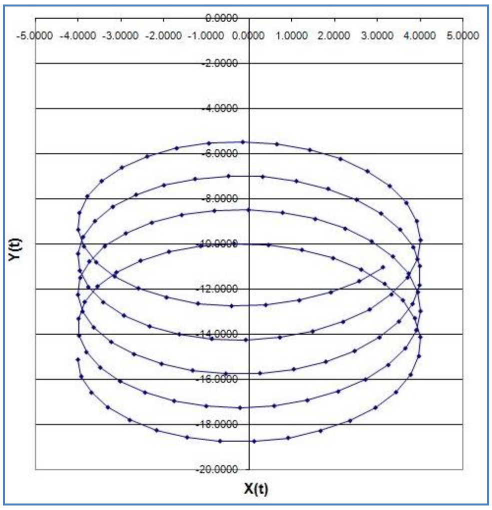


Fig. 6. Representation of the trochoidal toolpath milling

After obtaining the toolpath, a conversion of the results to a point file is performed (Table 3).
The purpose of the points (coordinates) file is to allow programming of the toolpath according to ISO 6983 (which defines the principles of the G code).

Table 3. Calculation of coordinates

t=	x(t)=	y(t)=	Lignes N°	Coordinates
0	-4.0000	-14.7500	1	X-4.000 Y-14.750
0.2	-3.9203	-15.5128	2	X-3.920 Y-15.513
0.4	-3.6842	-16.2440	3	X-3.684 Y-16.244
0.6	-3.3013	-16.9131	4	X-3.301 Y-16.913
0.8	-2.7868	-17.4921	5	X-2.787 Y-17.492
1	-2.1612	-17.9567	6	X-2.161 Y-17.957
1.2	-1.4494	-18.2872	7	X-1.449 Y-18.287
1.4	-0.6799	-18.4690	8	X-0.680 Y-18.469
1.6	0.1168	-18.4936	9	X0.117 Y-18.494
1.8	0.9088	-18.3589	10	X0.909 Y-18.359
2	1.6646	-18.0689	11	X1.665 Y-18.069
2.2	2.3540	-17.6338	12	X2.354 Y-17.634

4. NEURAL NETWORK APPLICATION OF A FIVE-AXIS MACHINE

The advantages of using ANN technique is that they have a capacity to learn based on optimization of an appropriate error function and they have an excellent performance for approximation of nonlinear functions.
Multilayer perceptron neural networks have better ability than other techniques to solve various complex problems. MLP is adept of performing nonlinear mapping between the input and output space due to its large parallel interconnection between different layers and the nonlinear processing characteristics. MLP neural network's takes a multi-dimensional input, and then delivers it to the other neurons according to their weights.
This donates a scalar result at the output of a neuron. The transfer function of a Multilayer perceptron neural network, employs a learning process to create a relationship between output and input. For the activation input, a time function is needed [12,16].
A back-propagation algorithm with multi-layer perceptron is used for the present problem. The network is trained with data for a number of Cartesian positions and orientation of the end effectors.
The structure of the considered MLP network is shown in Figure 7. The output vector is presented to a hidden layer neuron in the network via the input neurons. Each neuron of a layer is connected to every neuron of the next layer. The network uses a supervising learning, in which an input is presented to the network along with the desired output and the weights are adjusted so that the network attempts to produce the desired output.

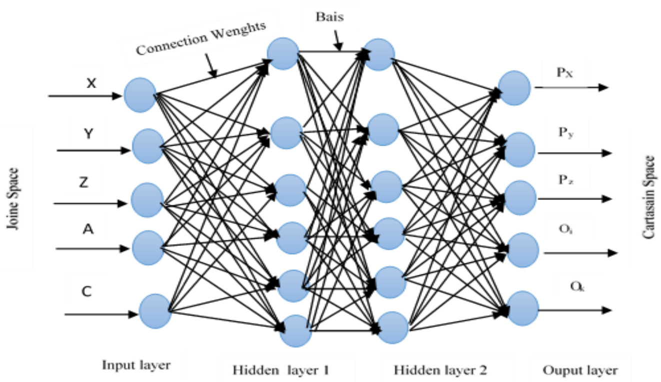


Fig. 7. Multi-layered perceptron neural network structure

4.1 Formulation of multilayer perceptron network

The MLP model involves an input layer, an output layer and generally one or more hidden layers [14].

Figure 8 shows the architecture of Multilayer Perceptron Network used for estimation of incremental joint space. It has an input layer of five neurons; one hidden layer of 25 neurons with sigmoid activation function (φ) defined by Eq. (7):

$$\varphi = \frac{1}{1+\exp(-y)} \qquad (7)$$

where y is the corresponding input.

For the output layer, a nonlinear activation function (sigmoid) is used in the implementation.

The approximating function Inverse Kinematic IK (ws), representing IK solution is defined by equations (8) and (9). The learning algorithm of MLP network includes the use of the input–output data to calculate the biases and weights. The training function updates the weights and bias values according to the gradient descent algorithm [15-16].Where:

$$IK(ws) = \phi\left[\sum_{j=1}^{m} A_j(ws)\,w_2(j) + b_2(1)\right] \qquad (8)$$

$$A_j(ws) = \varphi\left[\sum_{j=1}^{n} Ws(K)\,W_1(k,j) + b_i(j)\right] \qquad (9)$$

where ϕ is the nonlinear activation function at the output layer; Ws (k) is the kth element of input vector ws; M is number of hidden layer neurons.

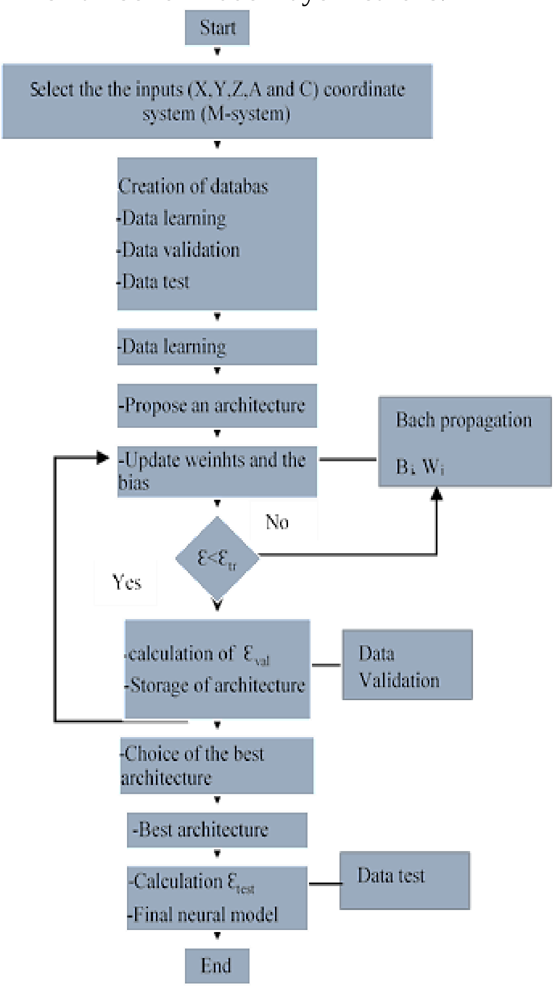


Fig. 8. Flow chart of the MLP neural network

The flow chart of the multilayer perceptron neural network in shown in Figure 8.

4.2 Neural Network Results and Discussions

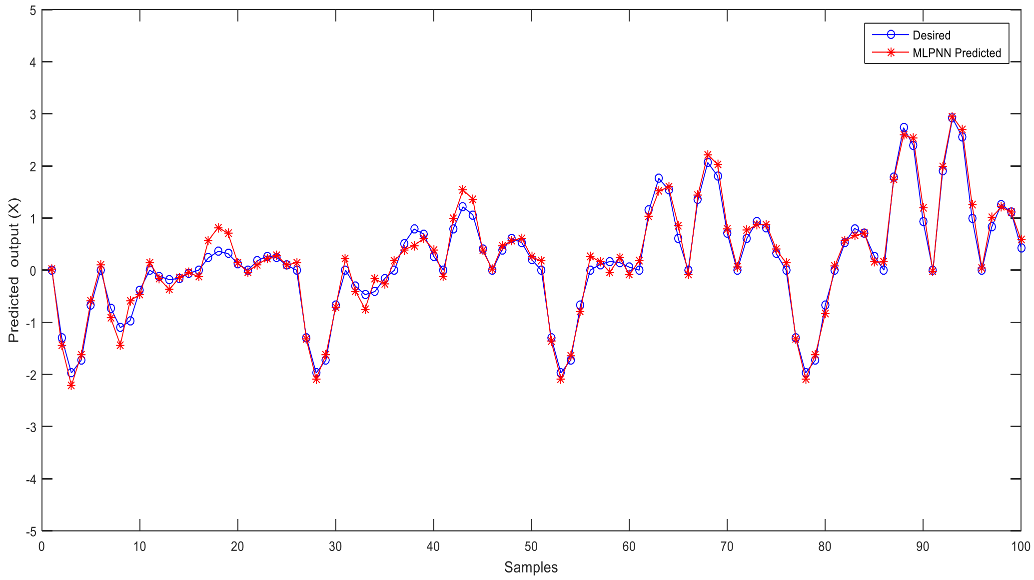
To validate the performance of proposed model based on Multilayer Perceptron Network for inverse kinematics problem, for predicting kinematic errors solutions, simulation studies are carried out using Matlab.

In this work, the training datasets are generated using equation (5). A dataset of 3125 points is first generated as per the formula for the input parameter (X, Y, Z, and A, C) coordinates in (mm). This dataset is randomly divided into training, validation, and test sets, 1042 data points is used as training data, 1040 is used for testing the MLP, and the remaining is used for the validation set. Back-propagation technique is employed for training the MLP network and for updating the desired weights. The MLP formulation is a generalized one and might be utilized for the solution of forward and inverse kinematics problem of manipulator of any configuration.

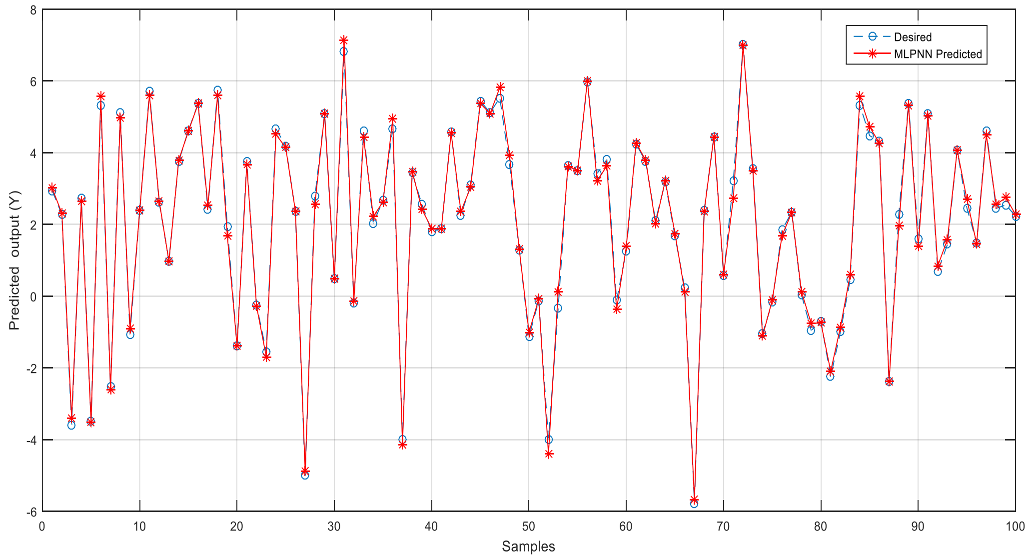
However, inverse kinematics configuration is considered in this paper to illustrate the applicability of the MLP model. To test the stability of the models validation data or testing data is essential as discussed earlier 1042 data points were selected randomly for testing the MLP model. In the Figure 9 represents the performance of the networks which was measured as the difference between desired and actual system output.

To drive the machine tool to track a chosen trajectory, it will be required to split the path into small portions, and to move the machine tool over all intermediate points. To accomplish this task at each intermediate location, the machine tool’s IK equations are solved, a set of joint variables is calculated, and the controller is directed to drive the machine tool to the next segment. When all segments are completed, Figure 9.a to 9.c shows the analytical trajectory tracking for the five-axis over the X, Y, Z, Coordinates of the global coordinates system for both of the network compared to each other verses the desired trajectory.

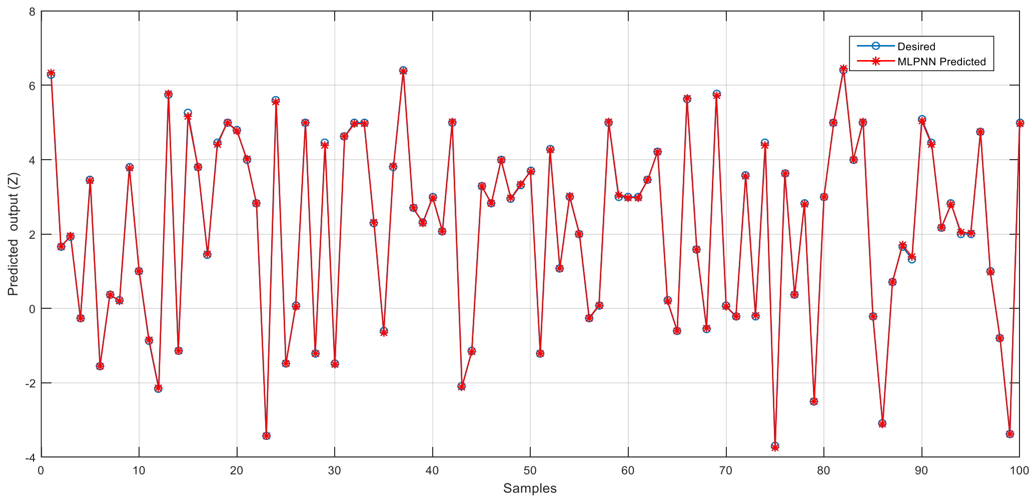
Figure 10, shows the best model selected using cross validation technique for the input coordinate X-axis, in which the error of validation is plotted on the y-axis and the number of iteration is plotted on the x-axis. Since we carried out and heuristic search in order to find the best number of neurons in the first and second hidden layers and the values of momentum and learning rate we used a new variable, which is the number of iteration, to count the iterations made by the algorithm to tune the parameters of the MLP model.



(a) Desired and Predicted values of X- axis



(b) Desired and Predicted values of Y-axis



(c) Desired and Predicted values of Z-axis

Fig. 9. Graph for matching of desired and predicted values of all input dataset

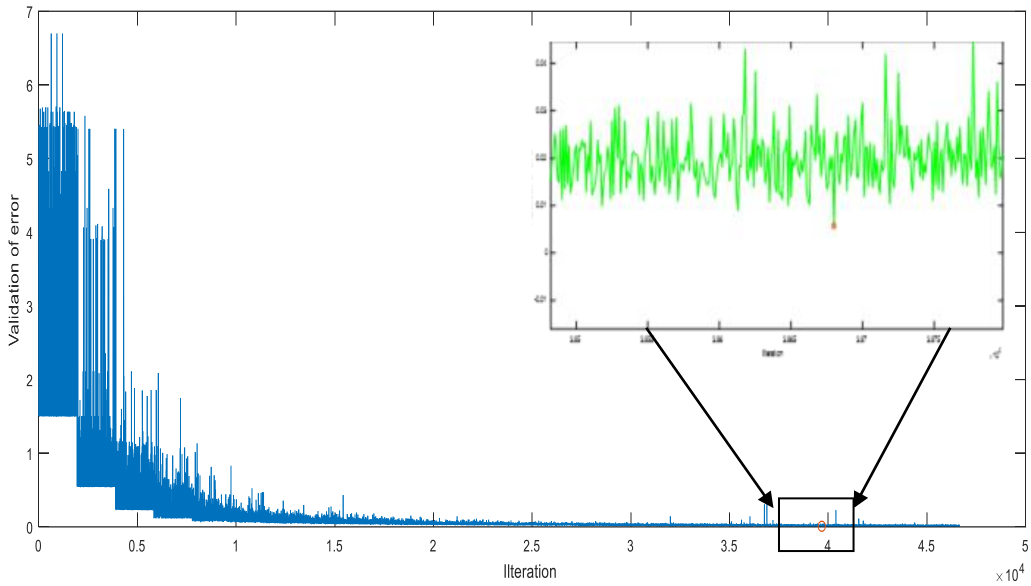


Fig. 10. Error of validation the X-axis

The best model corresponds to the lowest validation error (0.0052). In this case we find a number of neurons of 25 and 11 for the tow hidden layers respectively, and a momentum value of 0.3000 and a learning rate of 0.1100. The best MLP model uses the selected parameters in order to calculate the test error (0.0053). The results of the other input parameters are illustrated in table 5.

Table 5. The obtained MLP parameters

		X-axis	Y-axis	Z-axis
Dataset	Training set	1042	1042	1042
	Test set	1042	1042	1042
	Validation set	348	348	348
Number of neurons	Input layer	5	5	5
	Hidden layer1	25	22	18
	Hidden layer2	11	11	16
	Output layer	1	1	1
Learning rate	LR	0.1100	0.7100	0.7100
Momentum parameter	MU	0.3000	0.8000	0.1000
MSE-validation		0.0052	0.0057	0.0052
MES-Test		0.0053	0.0053	0.0061

The results in Figure 9 and 10 shows that the idea of using a neural network has produced an excellent approximation of the inverse kinematics function.

4.3 Machining of the workpiece tests

In order to validate the analytical engagement model, experimental studies were performed. After checking the curvature of the path and obtaining the NC program, the workpiece is machined (Figure 11).

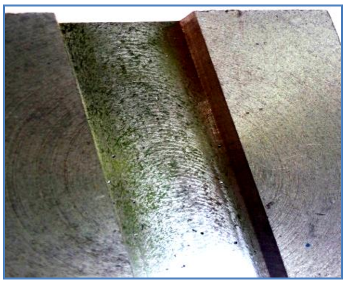


Fig. 11. Test piece of trochoidal machining of a groove

Implementation of emergent machining strategies type trochoidal strategy not only avoids machining situations in full milling, but also, to ensure a control of the instantaneous radial engagement.

This strategy is therefore well suited to machining situations for which the tool / material pair strongly restricts the choice of cutting conditions.

Finally, the machining of this piece tests is interesting for the machining of other forms.

5. CONCLUSIONS

This work presents an approach for predicting kinematic errors solutions by the generation of an optimal high-performance precision contour algorithm for five-axis machines.

The method is based on solving inverse kinematics using artificial neural network. A new design of multi-layer perceptron networks have been proposed for the optimization of kinematic errors of the five-axis. The problem of error value (mean square error) is nearly zero, which is very much acceptable when compared to the precision figures and repeatability error values of any typical manipulator.

From the present study, it is observed that the MLP gives minimum mean square error for resolution and joints variables as a performance index. This artificial neural network based joint position and rotation prediction model can be a useful tool for the production engineers to estimate the motion of the manipulator accurately.

The stability of trochoidal milling process is predicted by obtaining best model selected using cross validation technique for the input coordinate X-Y-Z axis made by the algorithm of the MLP model. The method is based on solving inverse kinematics using artificial neural network.

Based on the proposed stability model, a trochoidal step optimization strategy is developed to improve the machining efficiency of trochoidal milling under other parameters in a given situation. Cutting experiments are performed with part test to show the effectiveness of the proposed trochoidal milling stability model. Finally, simulations are adopted to illustrate the optimization strategy.

6. REFERENCES

1. Lee, R.S. and Lin, Y.H., (2012). *Applying bidirectional kinematics to assembly error analysis for five-axis machine tools with general orthogonal configuration*, Int J Adv Manuf Technol., 62(9-12), 1261-72.
2. Zhang, Z., Liu, Z., Cheng, Q., and al., (2017). *An approach of comprehensive error modeling and accuracy allocation for the improvement of reliability and optimization of cost of a multi-axis NC machine tool*, Int J Adv Manuf Technol., 89(1-4), 561-579.
3. She, C.H., and Chang, C.C., (2007). *Design of a generic five-axis postprocessor based on generalized kinematics model of machine tool*, Int J Mach Tool Manu, 47(3), 537-545.
4. Bohez, E. LJ., (2002). *Five-axis milling machine tool kinematic chain design and analysis*, Int J Mach Tool Manu, 42(4), 505-520.
5. Farouki, R. T., HAN, Chang Y., and LI, S., (2014). *Inverse kinematics for optimal tool orientation control in 5-axis CNC machining*, Comput Aided Geom Des, 31(1), 13-26.
6. Moon, S.K., Moon, Y.M., Kota, S., and Landers, R.G., (2001). *Screw theory based metrology for design and error compensation of machine tools*, In: *Proceedings of DETC*, Vol. 1, 697-707.
7. Xu, H.-Y., Hu, L., Hon-Yuen, T., et al., (2013). *A novel kinematic model for five-axis machine tools and its CNC applications*, Int J Adv Manuf Technol., 67(5-7), 1297–1307.
8. Sorby, K., (2007). *Inverse kinematics of five-axis machines near singular configurations*, Int J Mach Tool Manu, 47(2), 299-306.
9. Lamikiz, A., Lopez De Lacalle, L.N., Ocerin, O., Díez, D. and Maidagan, E. (2008). *The Denavit and Hartenberg approach applied to evaluate the consequences in the tool tip position of geometrical errors in five-axis milling centres*, Int J Adv Manuf Technol., 37(1), 122-139.
10. Yang, J., and Yusuf A., (2015). *A generalized on-line estimation and control of five-axis contouring errors of CNC machine tools*, Int J Mach Tool Manu, 88, 9-23.
11. Köker, R., and Çakar, T., (2016). *A neuro-genetic-simulated annealing approach to the inverse kinematics solution of robots: a simulation based study*, Eng Comput, 32(4), 553-565.
12.Panchanand, Jha, and Biswal, B.B., (2014). *A neural network approach for inverse kinematic of a SCARA manipulator*, International Journal of Robotics and Automation (IJRA), 3(1), 52-61.
13. Köker, R, Öz, C., Çakar, T. and Ekiz, H., (2004). *A study of neural network based inverse kinematics solution for a three-joint robot*, Robotics and autonomous systems, 49(3), 227-234.
14. Chiddarwar, S.S. and Babu, N.R., (2010). *Comparison of RBF and MLP neural networks to solve inverse kinematic problem for 6R serial robot by a fusion approach*, Engineering applications of artificial intelligence, 23(7), 1083-1092.
15. Lin, Yl, and Shen, Y., (2003). *Modelling of five-axis machine tool metrology models using the matrix summation approach*, Int J Adv Manuf Technol., 21(4), 243-248.
16. Feng, Y., Yao-nan, W. and Yi-min, Y, (2014). *Inverse kinematics solution for robot manipulator based on neural network under joint subspace*, Int J Comput Commun, 7(3), 459-472.
17. Rauch, M., Duc, E, Hascoet., J.Y., (2009). *Improving trochoidal tool paths generation and implementation using process constraints modeling*, Int J Mach Tool Manu, 49(5), 375-383.

Received: December 15, 2017 / Accepted: May 10, 2018 / Paper available online: June 20, 2018 © International Journal of Modern Manufacturing Technologies.