DATA DRIVEN MODELING OF LINEAR MOTION SYSTEM USING LEARNING ALGORITHM

H Ramesh[1*], Jeshuran Cornelius K S R[2], Mohamed ashwaque noor mohamed[3], Aadhavan P[4]

Abstract:
Linear motion control systems are used in a wide range of industrial applications, including machine tools, packaging, textiles, printing, and renewable energy. The main control elements of the linear motion control systems are the motion controller, servo drive, and encoder. The motion controller controls the servo drive coupled with the lead screw or ball-screw assembly in order to obtain the desired motion profile. The motion controllers need to be configured accurately with the parameters of motor revolution, load revolution, and length unit (LU) per load revolution in order to achieve the desired results. The positional errors due to temperature, vibration, backlash, and encoder measurement also affect the accuracy of the linear motion system. In this paper, the proposed experimental data-based modelling of the linear motion system using learning algorithms help the user to configure the motion parameters easily and also reduce the positional errors due to mechanical and measurement error factors. Learning algorithms such as the neural network (NN) and machine learning (ML) algorithms are used to compare and formulate the best model that gives the least root mean squared error (RMSE). The comparison and experimental validation show that the Gaussian process regression (GPR) model outperforms all other models.

Keywords: Lead Screw Assembly, Levenbergan-Marquardt (LM) Algorithm, Long Short-Term Memory Algorithm, Simotion Scout Software.

[1*,2,3,4]Department of Mechatronics Engineering, Thiagarajar College of Engineering, Madurai 625 015, Tamil Nadu, India

***Corresponding Author:** H Ramesh
*Department of Mechatronics Engineering, Thiagarajar College of Engineering, Madurai 625 015, Tamil Nadu, India

DOI: - 10.53555/ecb/2024.13.02.04

1. Introduction:

Automation systems have been introduced in production, textiles, printing and medical industries due to technological advancements like Machine Learning, artificial Intelligence and robotics. The main objectives of automation systems are to increase productivity, control production costs, improve the quality of the product, reduce time, and speed up the operation. Motion control systems are employed in the automation of a machine or a process. The motion control system consists of a controller, a servo drive, a servo motor, a lead screw or ball screw assembly, and an encoder. The block diagram of a linear motion control system is given in Fig.1 [1].

The motion controllers are programmed using programming languages such as Ladder logic (LD), function block diagram (FBD), Structured Text (ST), and Motion Control Chart (MCC) in order to obtain the desired motion profile in a linear motion system. The motion parameters such as position, velocity, acceleration, deceleration, and torque are configured in the controller in order to generate a motion profile [2]. The servo drive actuates the servo motor coupled with the lead/ball screw assembly based on the generated motion profile from the controller. The rotary or linear encoder coupled with the motor or lead screw assembly provides the feedback signal to the controller and servo drive for obtaining the desired motion profile in the motion system. The hardware used for motion control is classified as controller-based, PC-based, and drive-based [3].

The parameters such as the motor revolution, the load revolution, and the Length Unit (LU) per load revolution need to be properly configured in the motion control software based on the gear ratio, lead screw parameters, and type of encoder [4].

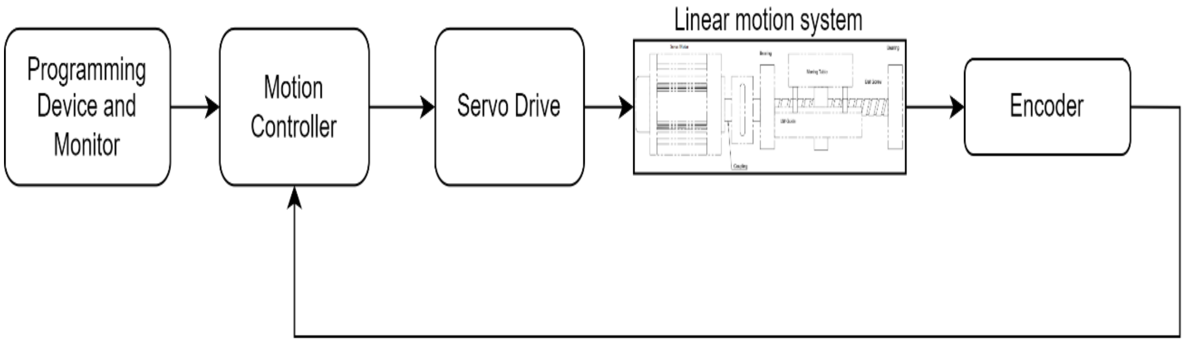


Fig. 1: Block Diagram of Linear Motion Control System

The position and velocity parameters are given as LU format in motion functions. There will be a relationship between the motion parameters given in LU format in motion functions and the motion parameters actually measured at the lead screw. As a result, in order to program the motion controllers for different positions and velocity at the lead screw assembly, the user must calculate the motion parameters in LU format each time. Feed screw pitch and torsion errors [5], temperature-induced errors [6, 7], and encoder measurement errors [8] all cause positional errors in lead screw or ball screw assemblies. This error leads to the variation between the actual motion parameters measured at the lead screw and the motion parameters given in the motion functions. The error compensation needs to be added to the motion parameters to achieve the efficient operation of the motion control system. Hence, user expertise is required in order to establish the relationship between the controller input and lead screw output parameters, and also for error compensation due to mechanical and measurement errors.

Artificial neural networks (ANN) are computational methods that are inspired by the neurological systems of animals. It's a machine learning algorithm whose model is based on the neurons of humans. Human brains are made up of millions of neurons that transmit and receive chemical and electrical impulses in a pattern. The optimization procedure finds the best solution by establishing the relationship between input and output. According to literature studies, artificial neural networks are effectively utilized for constructing the model by using the machining input and output parameters.

Shiba Narayan Sahu et al. (2018) proposed an ANN model integrated with NSGA-II for EDM machine using the process parameters of material removal rate and tool wear rate obtained through experiments [9]. Azlan Mohd Zain et al. (2012) developed an ANN model integrated with GA for obtaining the minimum value of machining performance parameters [10]. Chaki et al. (2018) proposed an integrated ANN-NSGA2 model that optimizes the surface roughness and material removal rate for laser cutting machine using a Bayesian regularization algorithm [11]. Amit kumar jain et al. (2016) proposed a data driven model using feed forward back propagation ANN model for prognostics of cutting tools [12]. Hung-Wei Chiu (2017) proposed ANFIS model for predicting machining accuracy and surface quality for CNC machine tools [13]. Girish Kant (2014) compared the results of ANN and support vector regression (SVR) by modeling the power consumption in the machining process [14].

Zaiwu Mei et al. (2019) proposed a data driven model using back propagation (BP) ANN for feed systems of machine tools [15]. Neelima Sharma et al. (2020) compared the effectiveness of different machine learning algorithms for automatic programming of computer numerical control (CNC) machine for machining different types of holes[16].Kim et al(2018) reviewed the use of machine learning algorithms for different machining process used in the machining industry[17].

Markus Brillinger (2021) proposed a real-data driven prediction of energy consumption in CNC machining using a random forest machine learning algorithm[18]. The machine learning algorithms such as support vector machine (SVM) and Bayesian method are used in industrial process control for modeling soft sensors [19]. Li Ai et al. used random forest algorithm for detecting acoustic emission employed for monitoring flight safety [20].

In this paper, the data-driven model of the controller-based lead screw assembly was created using the artificial neural networks, and machine learning algorithms. The best model that gives the minimum root mean square error (RMSE) is evaluated using the ANOVA main effects plot. The model that establishes the relationship between the motion parameters given in motion functions and actual motion parameters measured at a lead screw assembly helps the user to program motion controllers easily and also reduces the error that arises due to the mechanical, thermal, and encoder parameter variations. In this paper, the data-driven modeling of a siemens simotion based linear motion system is proposed for easy parameter configuration and error compensation.

This paper is organized as follows: Section 2 discusses the experimental data collection in the linear motion system. Section 3 is about the modeling of a linear motion system using the learning algorithms. Section 4 shows the comparative analysis of different learning algorithms used for modeling the linear motion system.

2. Experimental data collection in linear motion system

Linear Motion system consists of a motion controller where the motion programs are written for moving the lead screw assembly. The servo drive operates the servo motor coupled with a lead screw assembly based on the control signal from the motion controller. The encoder connected to the linear motion system provides feedback to the controller for correcting any deviation in motion parameters. The programming device and monitor are used to program the motion controller and monitor the motion parameters in real time. In this work, the Siemens Simotion controller based linear motion system which consists of the motion control, the PLC functionalities, and the technology functions is modelled by using the set of data obtained through a series of experiments [21]. Simotion based linear motion system consists of a Simotion D410 single-axis controller connected to a Sinamics S120 servo drive that drives the Simotics-S servomotor.

The Motion Control Program for the Lead Screw was performed using Simotion Scout Software. In Simotion-Scout Software, a project is created by adding D410-2 DP/PN Simotion device. Then the profinet interface is created between the D-410 controller and the Sinamics S-120 drive. The axis, the drive, the motor, and the encoder parameters are configured in the software.

Fig.2 shows the configuration of drive unit mechanics in LU format in Scout software. The LU per load revolution, the load revolutions, and motor revolutions need to be properly configured in this configuration block in order to achieve accurate positioning of linear systems. A Motion Control Program was developed by inserting the motion control blocks such as mc_power, mc_home, mc_moverelative, and mc_moveabsolute .As shown in Fig.3a and 3b, 400 sets of data were collected by repeatedly moving the lead screw assembly in various positions and velocities.

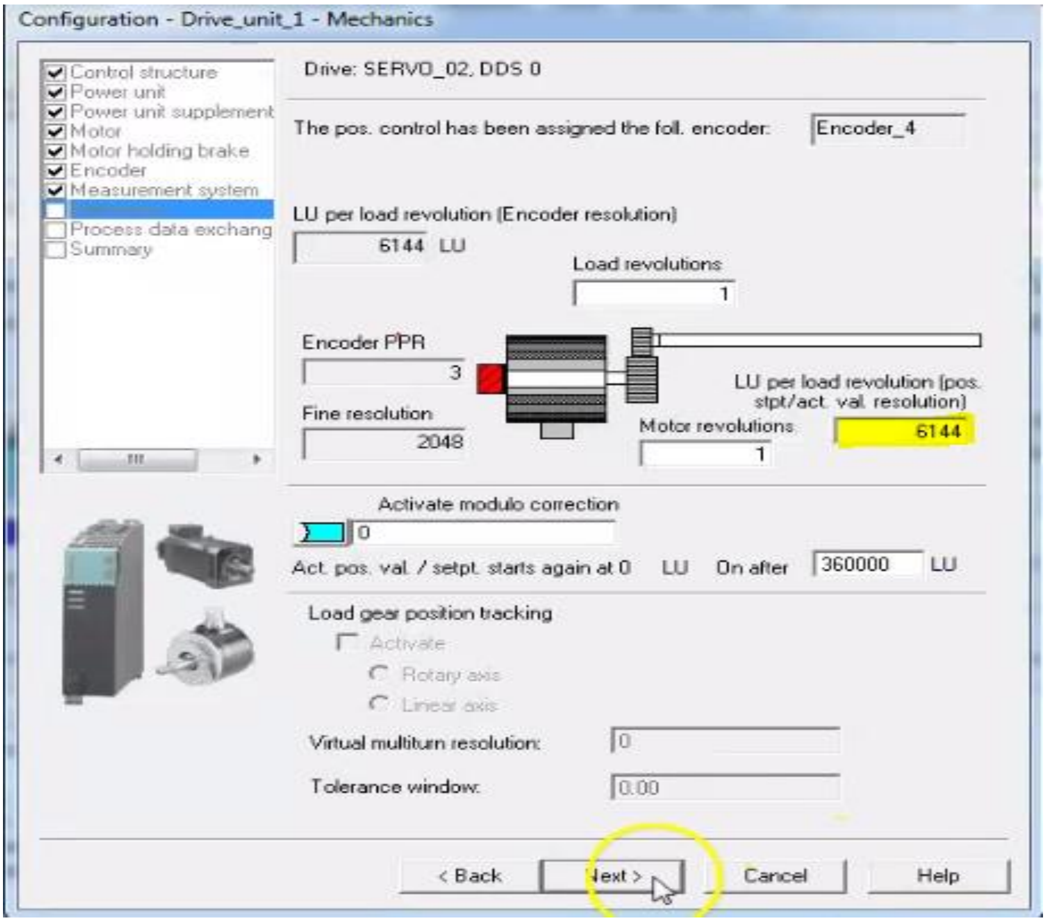


Fig. 2 Configuration block of Linear Motion Control System

(a) (b)

Fig. 3. (a) Position reached by the Lead Screw from Start Position (0 LU) to Relative Position (200 LU) (blue arrow) Home Position reached by the Lead Screw (blue arrow)

3. Modelling of linear motion system using Learning algorithms

The 400 sets of experimental data collected using the hardware setup are prepared for training, testing and validation. Fig.4 shows the flow chart of the methodology used in this work for developing optimal learning model of linear motion control system. For the learning models input data are given as position in LU units and velocity in mm/sec, and the output data are given as Distance moves and time taken respectively. The ANN models, regression models are trained, tested and validated using the prepared data. The models that give the least RMSE value are evaluated using the ANOVA main effects plot. The best model is exported and validated using the real time experimental data set.

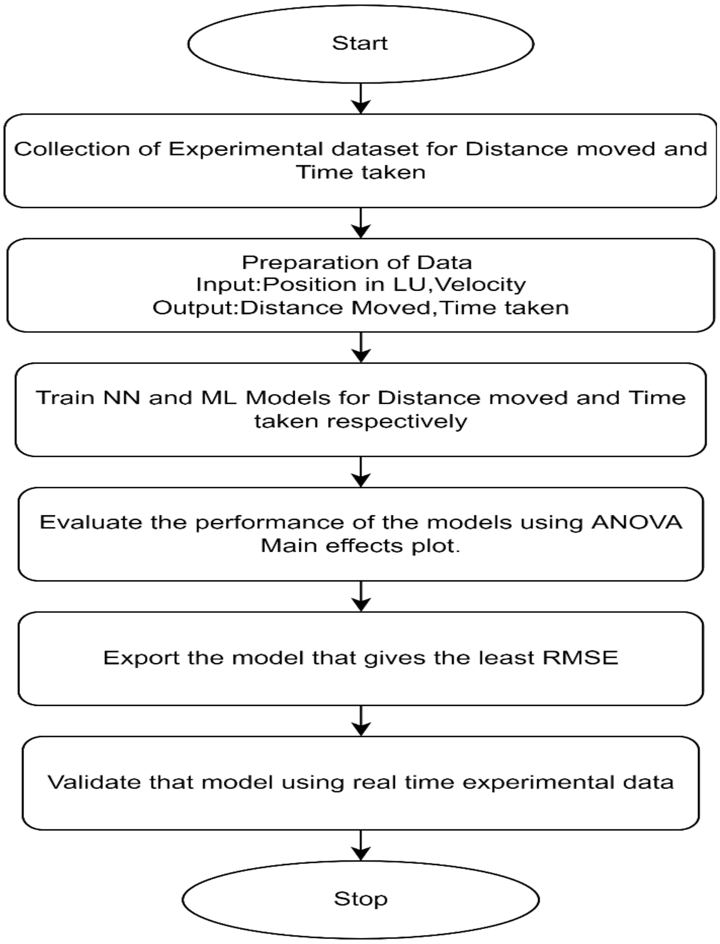


Fig.4 Flow chart of Methodology used to develop optimal learning Model

3.1ANN algorithm-based modelling:

A two-layered feed forward network with a hidden and output layer is used to create the model by fitting the input and output data of the linear motion system. The trained ANN model for the simotion-based linear motion system was developed using the Levenberg-Marquardt (LM) technique, the Bayesian Regularization (BR) algorithm, and the Scaled Conjugate Gradient (SCG) algorithm. The LM algorithm is the fastest back propagation algorithm [22] in the neural network toolbox of Matlab. It requires more memory but less time. Training of these neural networks stops when generalization starts improving. The BR model minimizes the linear combination of squared errors and weights. The good generalization qualities are obtained by modifying the linear combination during the training of the networks. It requires more time but can result in good generalization for difficult or small datasets. The SCG model trains the network until its weights, net input, and transfer function achieve derivatives. It requires less memory. Training automatically stops when the generalization stops improving.

The 400 data sets collected through the experimental setup are used to train and test the model developed using different training algorithms and hidden neurons as given in the Table.1.The ANN model is developed using the input parameters of position and velocity given to the motion blocks and the output parameters of distance moved and time taken, measured at the linear motion system. The root mean squared error (RMSE) for different combinations of training algorithm and number of neurons is used to select the best model that gives the least RMSE. The main effects plot is used to identify the best model which gives the least RMSE for both the distance moved and the time taken. The selected model is validated for 10 sets of real-time data using the simotion-based linear motion system.

Table.1. ANN model parameters

Training algorithm	Levenberg Marquardt	Bayesian Regularization	Scaled Conjugate Gradient	-	-
No of hidden neurons	4	8	12	16	20

3.2 Machine learning algorithm-based modelling:

The machine learning algorithm learns the information directly from the data using computational methods without relying on the predetermined model. It is classified in to supervised and un supervised learning algorithms. The supervised learning algorithm predicts the model using the input and output data. It creates the machine learning model using the regression and classification techniques. Regression learning techniques are used to predict the output variable based on the one or more predictor variables. The MATLAB machine learning tool box consists of models of regression decision tree, support vector machine (SVM), and Gaussian process regression models.

The regression tree algorithm divides the given data set into smaller groups before fitting a simple model to each subgroup. On the basis of the different predictors, successive binary partitions are used to partition the data. The variable partitioning is done in a top-down greedy manner. The average response values (y) for all observations in that subgroup are used to calculate the constant (c) to forecast. The model looks for every distinct value of every input variable to find the predictor and split value that divides data in to different regions(R) in order to minimize the over-all root mean squared error as given in Eqn.1

$$minimize\left\{RMSE = \sqrt{\sum_{i\epsilon R_1}(y_i - c_1)^2 + \sum_{i\epsilon R_2}(y_i - c_2)^2 + \cdots + \sum_{i\epsilon R_n}(y_i - c_n)^2}\right\} \text{-----(1)}$$

The goal of a SVM algorithm is to find a hyperplane in a n-dimensional space that categorizes data points clearly. Support Vectors are the data points on either side of the hyperplane that are closest to the hyperplane. These have an effect on the hyperplane's position and orientation, and so aid in the construction of the SVM.A kernel is a set of mathematical functions converts the input data into a required form in order to find the hyperplane in the higher dimensional space. The most commonly used kernels in SVM are Linear, sigmoid, Polynomial and Radial Basis Function (RBF). The support vector regression uses hyperplane and decision boundary to find the best fit that has maximum number of data points. If the equation of the hyper plane is

$$Y = wx + b \qquad \text{------(2)}$$

Then the decision boundary equations become

$$wx + b = +\sigma \;\; ; \; wx + b = -\sigma \qquad \text{------(3)}$$

Where $+\sigma$ and $-\sigma$ are decision boundaries drawn parallel to hyperplane.
As a result, every hyperplane that fulfils the state vector regression must satisfy the following conditions:

$$-\sigma < Y - wx + b < +\sigma \qquad \text{-----(4)}$$

Gaussian process regression defines a gaussian process prior, f(w) consisting of mean function, m(x), and covariance function,k(x,x') as given in Eqn.5

$$f(w) \sim GP(m(x), k(x, x')) \qquad \text{------(5)}$$

The posterior distribution f(w|y,x) is calculated using the bayesian probability distribution given by:

$$f(w|y,x) = \frac{f(y|x,w)f(w)}{f(y|x)} \qquad \text{-------(6)}$$

Where $f(y|x,w)$ is likelihood and $f(y|x)$ is the marginal likelihood.
The optimizable form of above algorithms that adjust its hyper parameters for obtaining a minimum RMSE are used to model the linear motion system. The total data is divided into 80% for training and 20% for testing. The position and velocity is given as input variable, and the distance moved and the time taken is considered as output variable. The model which gives the less RMSE in both distance moved and time taken is selected using the main effects plot.

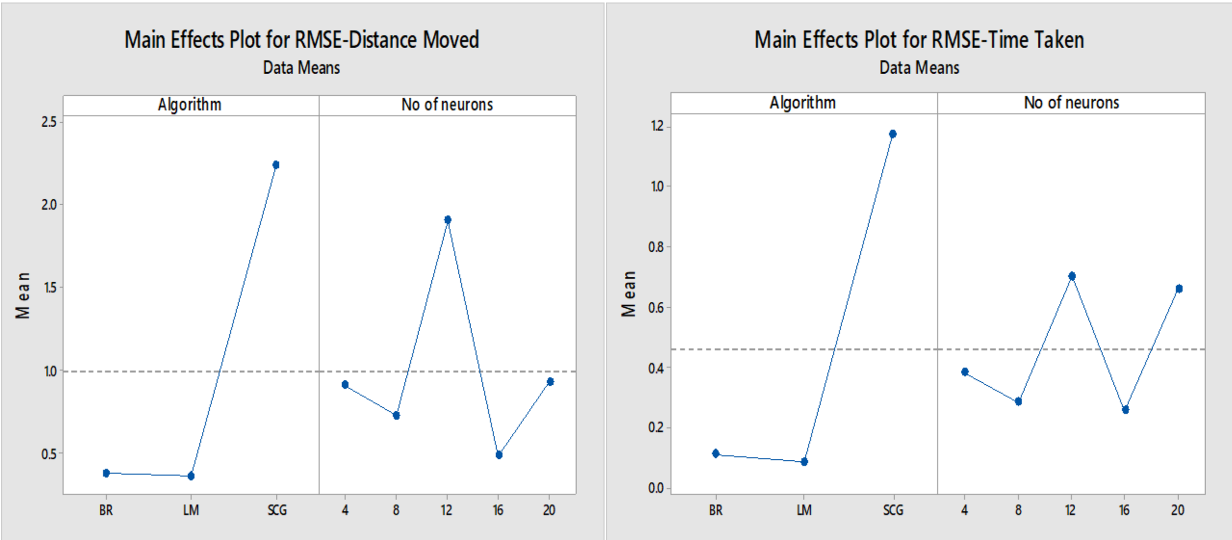
4 Results and Discussion:

The simotion controller based linear motion system is modeled using the 400 sets of data collected through a series of experiments with a position range from 0 to 650 LU and velocities of 50 and 70 mm/sec. Initially, the NN fitting tool is used to model the data. The LM, BR, and SCG algorithms are used to train the model with the hidden neurons of 4, 8, 12, 16, and 20. The input data is given as position in LU format, the velocity in mm/sec, and the out response is trained and tested separately as the distance moved in mm and the time taken in sec. Table.2 shows the RMSE value for the different NN algorithms with different hidden neurons.

Table2: RMSE of the linear motion system test-output results for different NN algorithm

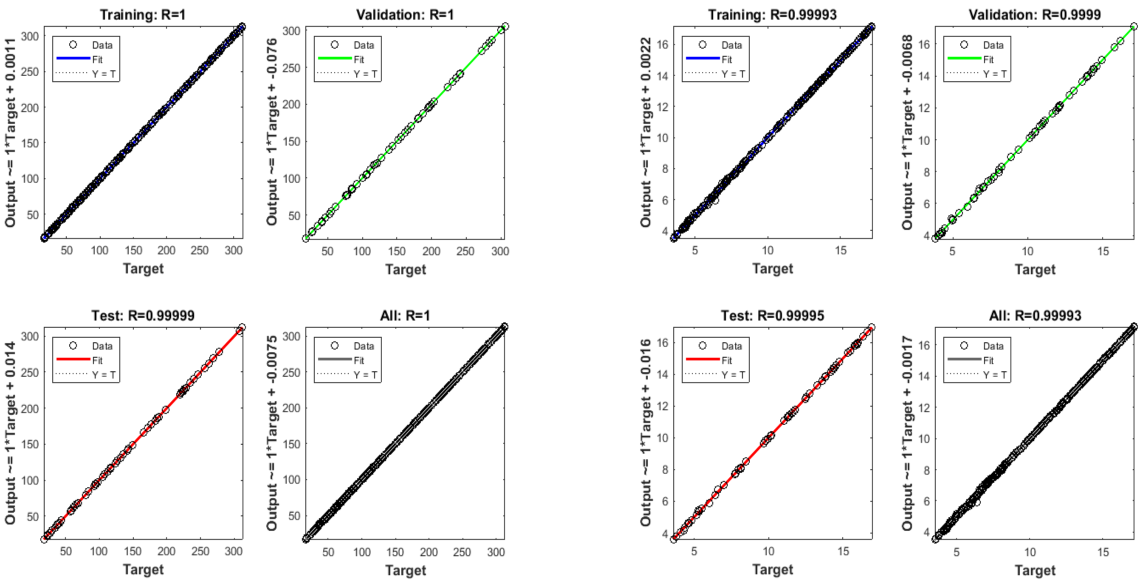
Training Method	LM					BR					SCG				
No.of Hidden Neurons	4	8	12	16	20	4	8	12	16	20	4	8	12	16	20
RMSE-Distance Moved	0.382	0.358	0.406	0.316	0.3503	0.371	0.377	0.381	0.3797	0.379	1.979	1.444	4.928	0.7611	2.068
RMSE-Time Taken	0.108	0.1345	0.093	0.0466	0.0542	0.100	0.111	0.116	0.1192	0.114	0.945	0.610	1.902	0.6088	1.8177

The best algorithm and its corresponding hidden neuron are found by the ANOVA main effects plot, as shown in Fig.5. The LM with the 16 hidden neurons shows the least RMSE in terms of distance moved as well as time taken. So, the LM 16 neuron model can be selected as the best model to predict the output responses for the given input parameters. The regression results of the training, validation, and testing of the LM-16 neuron model are shown in Fig.6.



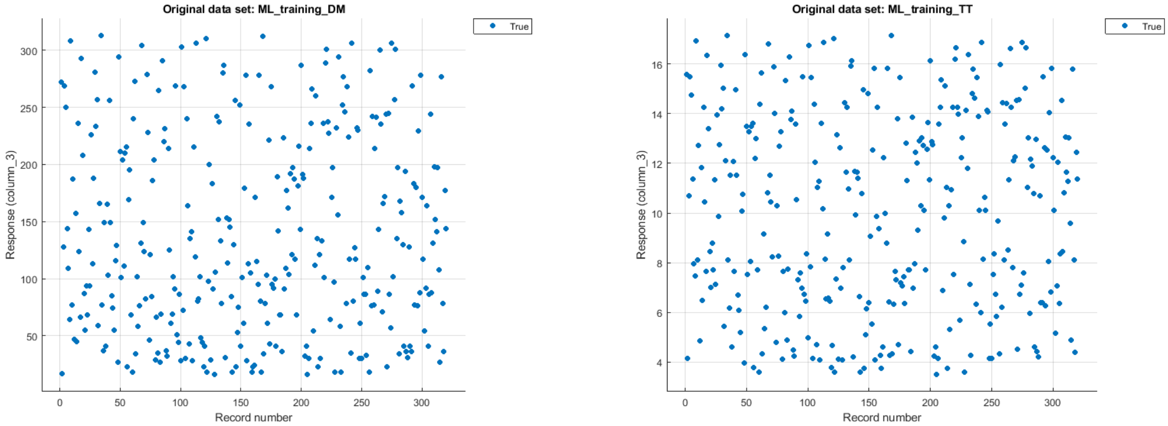
a b

Fig.5. Main effects plot of NN algorithms a. Distance Moved b. Time taken



a b

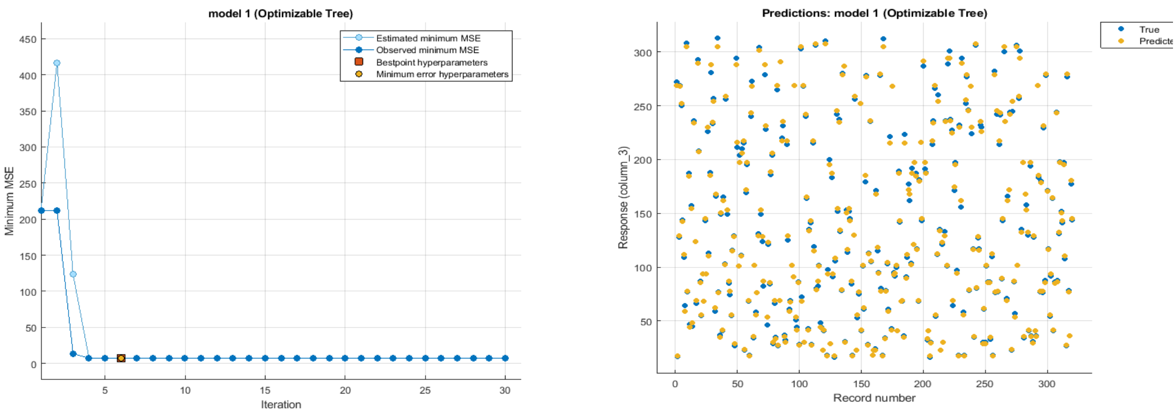
Fig.6. performance results of LM-16 neuron model for a. Distance Moved b. Time taken



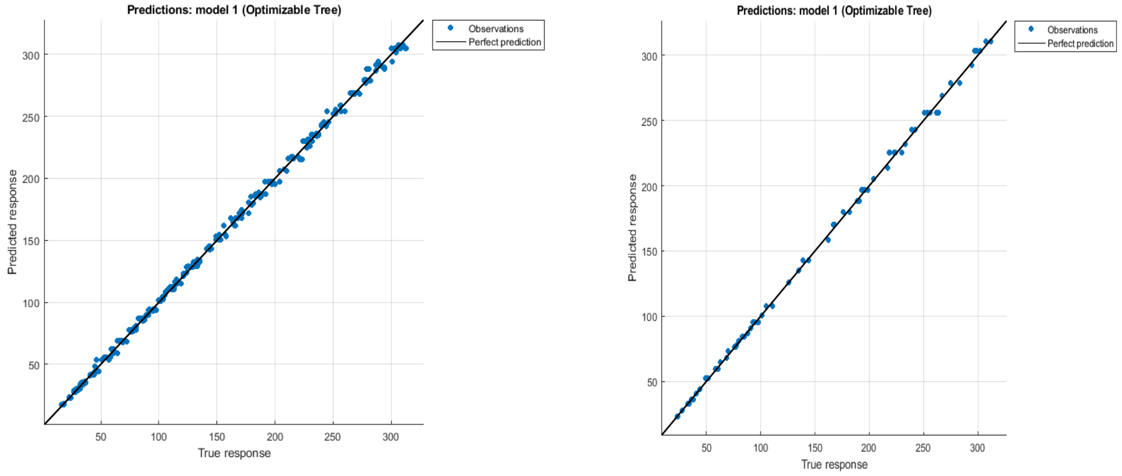
a **b**

Fig.7 .ML Training data set for a. Distance moved b. Time taken

ML algorithms such as optimizable Tree, SVM, and GPR are used to model the linear motion system using the Matlab. The training data of 80 % and testing data of 20% are used to model the linear motion system for the output response of distance moved and time taken as shown in Fig.7.The optimizing progress of the tree model for the linear motion system, the true and predicted response of training data, and testing data are given in Fig.8. The performance result of this model shows considerable variation between the predicted and true value. The optimizing progress of the SVM model for the linear motion system, the true and predicted response of the training data, and the testing data are given in Fig.9.This model shows tremendous improvement in the predicted result compared to the tree model. The performance result of the GPR algorithm is given in Fig.10.The GPR model provides the best prediction in both training and testing data compared to the tree and SVM models.

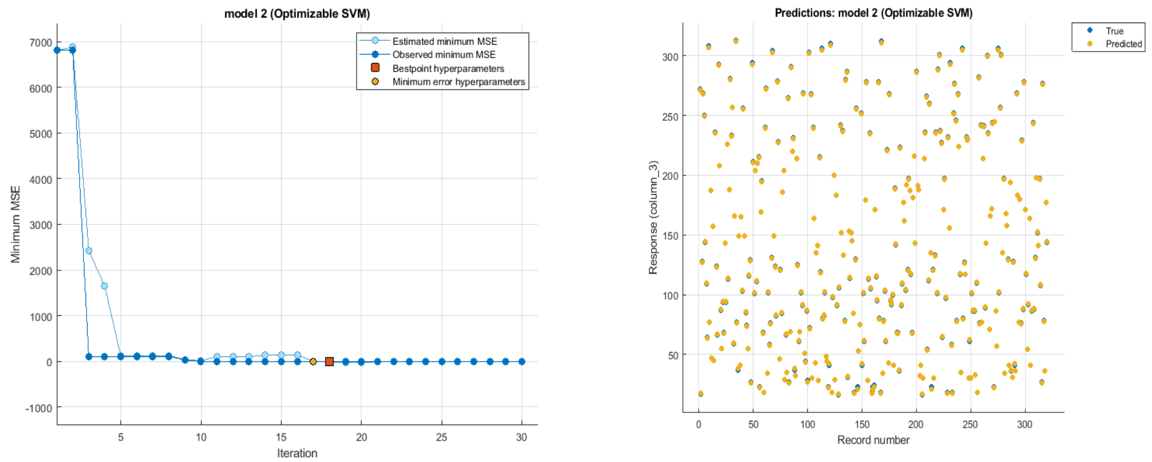


a **b**

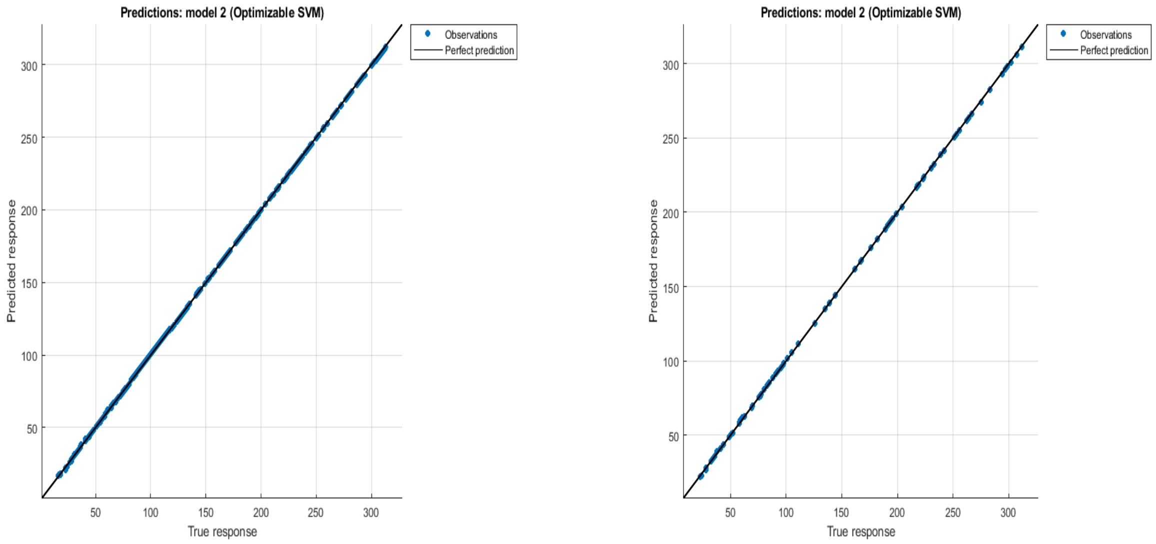


c **d**

Fig.8. Optimizable tree performance result a. optimizing tree model progress b. Response plot c. predicted vs actual plot for training d. predicted vs actual plot for training



a **b**



c **d**

Fig.9. Optimizable SVM performance result a. optimizing tree model progress b. Response plot c. predicted vs. actual plot for training d. predicted vs. actual plot for training

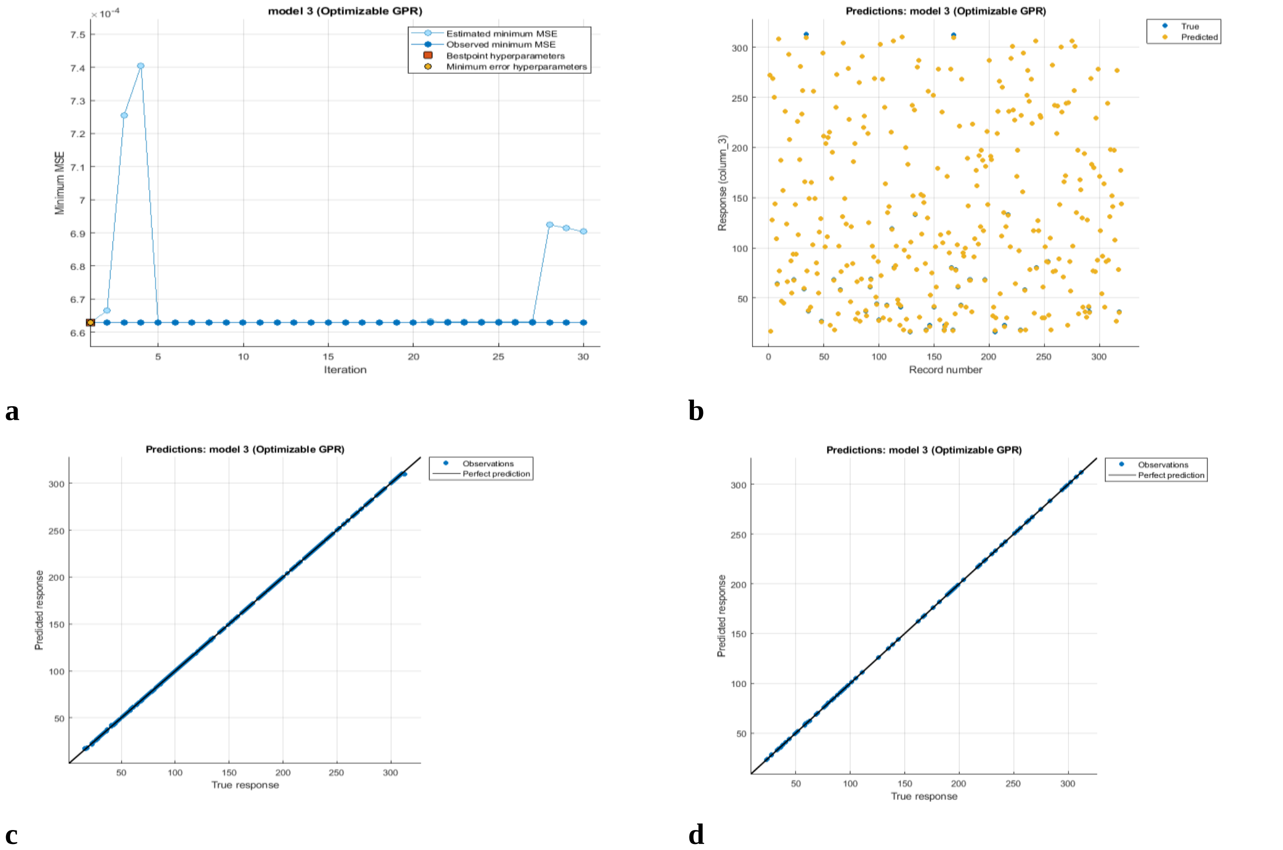
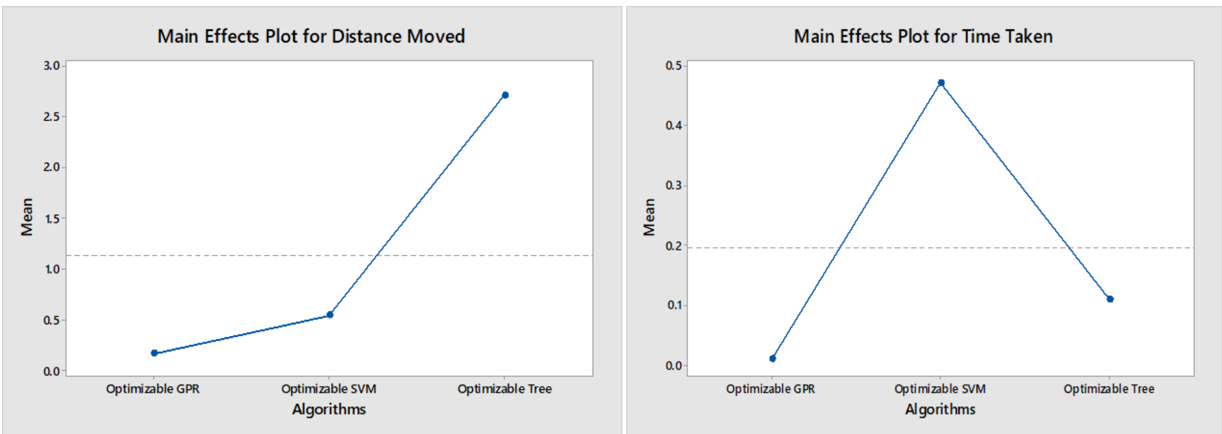


Fig.10. Optimizable GPR performance result a. optimizing tree model progress b. Response plot c. predicted vs. actual plot for training d. predicted vs. actual plot for training

The RMSE values for the training and testing data sets for the Tree, SVM, GPR and LSTM algorithm is tabulated in Table.3. The GPR algorithm shows a less RMSE value in both distance moved and time taken compared to the other three algorithms as shown in main effects plot given in Fig.11.The practical validation is done for the best performance model in NN, the LM model and the optimizable GPR model by conducting experiment with linear motion system with different position and velocity values as shown in Table.4.The GPR algorithm shows less RMSE compared to the LM algorithm in terms of distance moved and the time taken.

Table3: RMSE of the linear motion system output parameter for different ML algorithm

Output Response	**Optimizable Tree**		**Optimizable SVM**		**Optimizable GPR**	
	Training	**Testing**	**Training**	**Testing**	**Training**	**Testing**
Distance moved	2.672	2.7095	0.5545	0.5438	0.3310	0.1650
Time Taken	0.1683	0.1096	0.7710	0.4704	0.0258	0.0098



a b

Fig.11a) and b). Main effects plot of ML and DL algorithms

Table 4: Validation results of linear motion system using LM and GPR algorithm

Position in LU	Velocity in mm/s	Measured values		LM algorithm		GPR algorithm		LM Error		GPR Error	
		Distance Moved in mm(A)	Time Taken in sec(B)	Distance Moved in mm(A1)	Time Taken in sec(B1)	Distance Moved in mm(A2)	Time Taken in sec(B2)	Error Distance Moved(A-A1)	Error Time taken(B-B1)	Error Distance Moved (A-A2)	Error Time taken (B-B2)
538	50	272	15.59	271.996	15.586	272	15.593	0.004	0.00347	0	0.003
26	50	17	4.16	16.992	4.123	17.0528	4.1823	0.008	0.03686	0.0528	0.0223
248	50	128	10.69	127.956	10.688	128	10.689	0.0436	0.00159	0	0.0012
532	50	269	15.49	268.99	15.474	269	15.490	0.0093	0.0162	0	0.0003
494	50	250	14.75	250.008	14.751	250	14.749	0.0082	0.00127	0	0.0007
212	70	109	7.97	108.961	7.989	109	7.9693	0.0392	0.01851	0	0.0007
86	70	47	4.85	46.951	4.858	46.9999	4.8516	0.0491	0.00771	0.0001	0.0016
182	70	94	7.14	93.957	7.1506	94	7.1435	0.0428	0.01066	0	0.0035
112	70	59	5.44	59.351	5.4419	59.1247	5.4401	0.351	0.00193	0.1247	1E-04
74	70	41	4.62	40.866	4.6146	40.9277	4.6209	0.1337	0.00531	0.0723	0.0009
538	50	272	15.59	271.996	15.587	272	15.593	0.0689	0.01035	0.0249	0.00343
							RMSE	**0.2625**	**0.10174**	**0.1581**	**0.05856**

5. Conclusion:

The data driven modeling of the linear motion system using the NN algorithm and ML algorithm are investigated in this paper. The comparison results of the different algorithm done through the main effects plot and the experimental validation done for the two best models by importing from the matlab tool box.

The validation results show that the GPR model reduces the RMSE value about 50% than LM model in both distance moved and time taken. Hence, the GPR model can be used to configure the motion blocks of the Simotion based linear motion system easily and also to reduce the positional error due to the mechanical and measurement error factors.

7. Future Works:

In this paper, experimental data set of 400 with the velocity value of 50 mm/s and 70 mm/s are considered to model the system.

In future, this velocity range can be increased and many more data can be collected in order to train the model accurately. In this work ANN and ML models are considered for comparison. In future, deep learning models are also considered by using large dataset.

References:

1. Stephen J.O'Neil , *Motion Control Handbook*, Micro Mo Electronics,Inc,1998
2. *Motion Control NI-Motion User Manual*, 371242B-01, November 2005.
3. Motion Control System, *SIMOTION Equipment for Production machines manual*, Siemens Edition 2017.
4. *Siemens Sinamics S120 Drive Functions Function manual*, July 2007.
5. Hu Shi, Chi Ma, Jun Yang, Liang Zhao, Xuesong Mei, Guofang Gong(2015) "Investigation into effect of thermal expansion on thermally induced error of ball screw feed drive system of precision machine tools", *International Journal of Machine Tools and Manufacture*, Volume 97,2015,Pages 60-71,ISSN 0890-6955.
6. Liu, Kuo& Chen, Fei& Zhu, Tiejun& Wu, Yuliang& Sun, Mingjia,"Compensation for spindles axial thermal growth based on temperature variation on vertical machine tools", *Advances in Mechanical Engineering*, 2016,8.10.1177/1687814016657733.
7. Li, Yang & Zhang, Jun &Su, Dongxu & Zhou, Changxing& Zhao, Wanhua. (2018), Experiment-based thermal behavior research about the feed drive system with linear scale, *Advances in Mechanical Engineering*, 10.168 781401881235.10.1177/168781401881239.
8. F. Hu, X. Chen, N. Cai, Y. J. Lin, F. Zhang and H. Wang,(2018) Error analysis and compensation of an optical linear encoder, in *IET Science, Measurement & Technology*, vol. 12, no. 4, pp. 561-566, 7 2018.
9. Shiba Narayan Sahu and Narayan Chandra Nayak,(2018) Multi-objective optimisation of EDM process using ANN integrated with NSGA-II algorithm. *International journal of Manufacturing Technology and Management, Vol. 32, Nos. 4/5, 2018.*
10. Zain, Azlan & Haron, Habibollah & Sharif, Safian. (2012) Integrated ANN–GA for estimating the minimum value for machining performance. *International Journal of Production Research.* 50. 191-213. 10.1080 /00207543.2011.571454.
11. Sudipto Chaki & Ravi N. Bathe & Sujit Ghosal & G. Padmanabham,(2018)Multi-objective optimisation of pulsed Nd:YAG laser cutting process using integrated ANN–NSGAII model. *Journal of Intelligent*

Manufacturing, Springer, vol. 29(1), pages 175-190, January.

12. Amit kumar jain and Bhupesh kumar lad,(2016) Data Driven Models for Prognostics of High Speed Milling Cutters. *International Journal of Performability Engineering*, Vol. 12, No. 1, January 2016, pp. 03-12.
13. Hung-Wei Chiu, Ching-Hung Lee (2017) "Prediction of machining accuracy and surface quality for CNC machine tools using data driven approach" *Advances in Engineering Software*, 0965-9978,2017, http://dx.doi.org/ 10.1016/j.advengsoft.2017.07.008.
14. Girish Kant, Kuldip Singh Sangwan, (2014) Predictive Modeling for Power Consumption in Machining using Artificial Intelligence Techniques. *12th Global Conference on Sustainable Manufacturing*, 2212-8271, DOI:10.1016/j.procir.2014.07.072.
15. Zaiwu Mei , Jianwan Ding, Liping Chen , Ting Pi and Zaidao Mei,(2019) Hybrid Multi-Domain Analytical and Data-Driven Modeling for Feed Systems in Machine Tools, *symmetry 2019*, 11, 1156; DOI:10.3390/sy m11091156.
16. Sharma, Neelima & Chawla, V. & Ram, N.. (2020), Comparison of machine learning algorithms for the automatic programming of computer numerical control machine. *International Journal of Data and Network Science*. 4. 1-14. 10.5267/j.ijdns.2019.9.003.
17. Kim, Dong-Hyeon & Kim, Thomas & Wang, Xinlin & Kim, Mincheol & Quan, Yingjun & Oh, Jin & Min, Soo-Hong & Kim, Hyungjung & Bhandari, Binayak & Yang, Insoon & Ahn, Sung-Hoon. (2018).Smart Machining Process Using Machine Learning: A Review and Perspective on Machining Industry. *International Journal of Precision Engineering and Manufacturing-Green Technology*, 5. 555-568.10.1007/s40684-018-0057-y.
18. Markus Brillinger, Marcel Wuwer, Muaaz Abdul Hadi, Franz Haas(2021).Energy prediction for CNC machining with machine learning,*CIRP Journal of Manufacturing Science and Technology*,Volume 35,2021, Pages 715-723,ISSN 1755-5817, https://doi.org /10.1016/j.cirpj. 2021.07.014.
19. Yan, Weiwu & Shao, Huihe & Wang, Xiaofan. (2004). Soft Sensing Modeling Based on Support Vector Machine and Bayesian Model Selection. *Computers & Chemical Engineering*. 28. 1489-1498. 10.1016 /j.compchemeng.2003.11.004.
20. Li Ai ,Vafa Soltangharaei , Mahmoud Bayat ,Michel Van Tooren ,Paul Ziehl (2021) Detection of impact on aircraft composite structure using machine learning techniques, *Measurement Science and Technology,Vol.32,No.8,2021*
21. Daş E, Çağlar Başlamışlı S. (2021) Data-driven fixed-order H∞ controller synthesis in frequency domain: Closed-loop system approach. *Transactions of the Institute of Measurement and Control*. 2021;43 (5):1059-1067. DOI:10.1177/0142331219847741.
22. Alhanjouri, Mohammed. (2017). Speed Control of DC Motor using Artificial Neural Network. *International Journal of Science and Research (IJSR)*. 7. 2140 - 2148. 10.21275/ART 20172 035.
23. Hochreiter, S., and J. Schmidhuber.(1997) Long short-term memory. *Neural computation*. Vol. 9, Number 8, 1997, pp.1735–1780.