International Journal of Aquatic Science
ISSN: 2008-8019
Vol 12, Issue 02, 2021

System Identification Of Coupled Electric Drive Using Machine Learning Algorithms

Swetha R Kumar[1], Delina Rajkumari R D[2], D Jayaprasanth[3]

[1,2,3]*Department of Instrumentation and Control Systems Engineering, PSG College of Technology, Coimbatore,Tamil Nadu, India*

Email: [1]srk.ice@psgtech.ac.in, [2]delinadavid1997@gmail.com, [3]djp.ice@psgtech.ac.in

Abstract: System identification is behaviour of the system and hence nonlinear system identification is important. Machine learning which the application of artificial intelligence technique are programs that can learn and adapt to the changes without human interference. These are implemented in such a way that they try to find the relationship between the inputs and the target. Hence machine learning algorithms such as gradient descent based linear regression and neural networks are implemented for a coupled electric drive system. The simulation results are then analysed for their performance indices. the method of developing mathematical relationship between the input and output variables. Linear models cannot capture the non-linear

Keywords: Coupled electric drive; machine learning; linear regression; Gradient descent algorithm; Neural Networks

1. INTRODUCTION

The control community makes use of mathematical models intensively to design high-quality controllers as well as for the purpose of estimation. With the use of physical laws, these mathematical models are obtained using first principles approach, making use of detailed knowledge of the system. These are inefficient when it comes to system with non-linearity [3]. Hence data driven modelling is used where the relationship between input and output is obtained from data driven approach which includes constructing model from experimental data.[2] The traditional method of system identification can account for only local minimum but not global minimum. Also this proves better only for systems with known non linearity. Therefore machine learning approaches are used to obtain more accurate model[7].

Wei Xing Zheng proposed a bias eliminated least squares algorithm[8] to identify transfer function for a linear time invariant system. The main feature of this algorithm is that the transfer function co-efficient are estimated in such a way that there is no need to pre filter observed data.

Sajjad Ahmed Ghauri, Muhammad Farhan Sohail proposed a non-linear system identification method employing adaptive filters [1]. The work includes employing least mean squares, normalized least mean squares and recursive least mean squares method for the purpose of identifying the filter co-efficient.

Nidhil Wilfred K J, Sreeraj S, Vijay B proposed a method of nonlinear system identification using neural networks [4]. It involves back propagation approach. The method uses shallow neural network. The objectives of the work are to obtain non-linear ARX model of the system using Gradient descent and neural networks and to compare the models obtained through various algorithms by their performance indices such as mean square error and integral square error.

COUPLED ELECTRIC DRIVE SYSTEM

The CE8 coupled electric drives contains two electric motors that drive a pulley employing a flexible belt [5]. The pulley is held by a spring which results in lightly damped dynamic mode. The electric drives can be individually controlled thus allowing the tension and the speed of the belt to be controlled simultaneously. The drive control is symmetric around zero; hence both clockwise and counter clockwise movement is feasible. Here the focus is only on the speed control system. The reason is that the angular speed of the pulley is measured with a counter and this sensor is insensitive to the sign of the speed.

Following the sensor, two filters are applied which are analogue low pass filter and anti-aliasing filter. The spring and analogue low pass filter results in time constant thus generating dynamic effects. The analogue low pass filter has a quite limited effect on the output and may be neglected [6]. This system simulates the actual industrial problem in tension and speed control as they are used in paper mills, textile mills, strip metal production plants, magnetic tape drives etc. The setup of coupled electric drive system is shown in Fig 1.

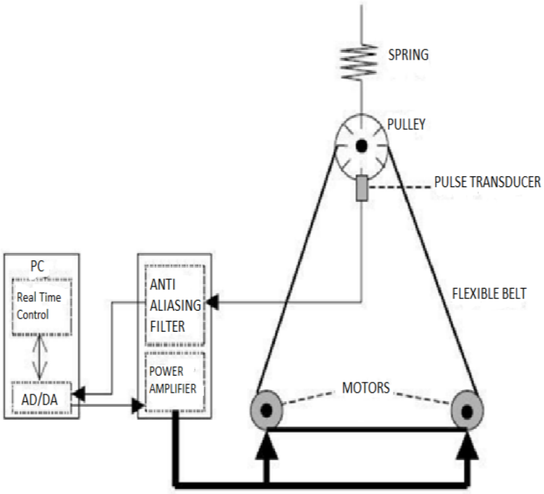

Fig 1 Coupled electric drive setup

The input signal was a PRBS with a clock period of 5 times the sampling period. The signal was switching between $+u_{prbs}$ and $-u_{prbs}$, resulting in the process changing the belt rotation direction frequently. Three open loop realizations were recorded for u_{prbs} 0.5, 1, 1.5. The data was collected with a sampling period of 20 ms.The 50 Hz sampling frequency is significantly higher than the bandwidth of the anti-aliasing filter. The durations of the measurement were 10 s, resulting in 500 input samples for each set of data.
The plot of input signal and output against time is shown in Fig 2a and Fig 2b respectively.

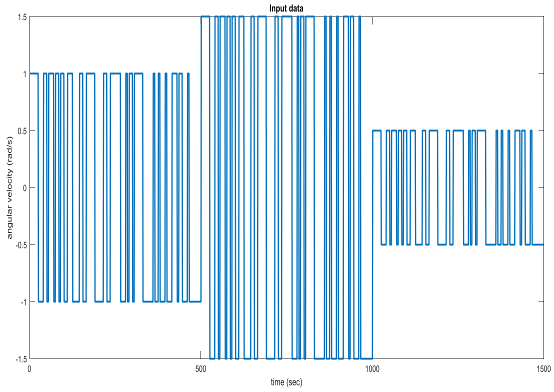

Fig 2a Input data

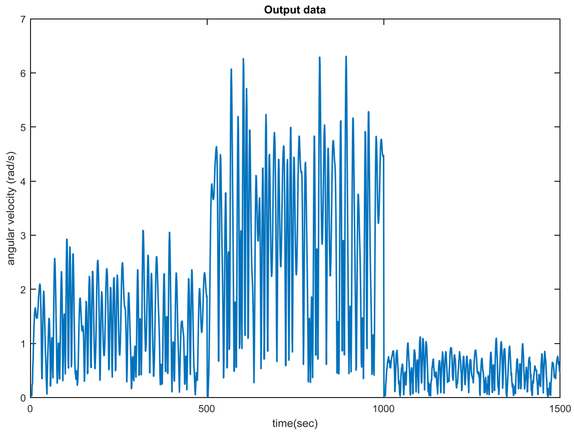

Fig 2b Output data

Hence a total of 1500 input-output data was obtained. These 1500 observations were used for training and validating the model. The sample data is shown in Table 1.

Table 1 Sample data

u_{prbs}	OUTPUT SIGNAL	u_{prbs}	OUTPUT SIGNAL	u_{prbs}	OUTPUT SIGNAL
1	0.031	-1.5	0.0179	0.5	0.6910
1	0.215	1.5	0.01221	0.5	0.7095
1	0.573	1.5	1.5397	0.5	0.8563
1	0.730	-1.5	1.5162	-0.5	0.8878
-1	1.037	-1.5	1.0189	-0.5	0.7040
-1	1.243	-1.5	0.9063	-0.5	0.6210
-1	1.428	1.5	0.8326	0.5	0.5521
1	1.556	-1.5	0.6741	0.5	0.5987
1	1.677	1.5	0.9179	0.5	0.6521
-1	1.995	1.5	1.2987	0.5	0.6678
-1	1.593	1.5	2.5152	-0.5	0.7190
-1	0.685	1.5	3.2154	-0.5	0.6975
-1	0.302	1.5	4.6149	-0.5	0.6410

IDENTIFICATION USING MACHINE LEARNING

System identification is a technique of building mathematical models of dynamical systems using batch or continuous measurements of the system's input and output signals. System Identification refers to the process of extracting information about a system from measured input-output data. The outcome is identification of model which may be static or dynamic, deterministic or stochastic, linear or nonlinear. These models can be used for the purpose of simulation, controller design, or analysis.
Different types of identification are as follows

White box: Model structure based on physical or natural laws.

Grey box: model structure with partial knowledge using first principles approach, the rest is reconstructed from data.

Black box: model structure and its parameters are unknown; they are estimated from input-output data.

The process of system identification requires to

- Measure the input and output data from the system in time or frequency domain.
- Selection of a model structure
- Apply an estimation method to estimate the tuneable parameters in the selected model.
- Evaluate the estimated model for its adequateness.

The method of building model from input-output data usually consists of three main elements – the data, set of candidate model and model estimation method.

The data

The input-output data that is used to select a model is the fundamental information source. To select the signals to be measured, to decide how the input should be configured, and to collect the data with appropriate sampling procedures will have a major impact on the quality of the resulting model.

Set of candidate model

For linear identification the choice of model sets is quite easy to grasp. In contrast, the choice of a model set for nonlinear identification is a major problem and offers a very rich range of possibilities. In any case, a model should be capable of producing a model output $\hat{y}$ (t) for the output at time t based on previous input-output measurements. This could be computed as a formal prediction of the output, or it can be based on other considerations. The set of candidate models is typically parameterized by a parameter vector, and the notation $\hat{y}$ (t|i) will be used for the model output corresponding to the model parameter.

The estimation method

With a given data set and model set, the identification task is to select that model that best describes the observed data. Most such estimation methods are based on a criterion of fit between the observed output y(t) and the model output $\hat{y}$(t|i), which can conceptually be written as

$$\hat{\theta} = argmin_{\theta} \sum_{t=1}^{N} ||y(t) - \hat{y}(t|\theta||^2 \qquad (1)$$

System identification procedure consists of the following four steps.
Experiment data – Collection of input-output data
Model structure- The model structure and specifications are to be selected.

Fit criterion- Select the model that fits the cost function
Validate criterion- Test the model for minimum cost function.
Cost function

To obtain a best fit, the model must be designed in such a way that the predicted value y is close to the actual output by the system. In other words the error between the predicted output of the model and the actual output of the system must be minimum. Hence the tuning parameters must be adjusted in such a way that the error is minimum. Thus the cost function can be formulated as

$$minimize\,\frac{1}{n}\sum_{i=1}^{n}(pred_i - y_i)^2 \qquad (2)$$

$$J = \frac{1}{n}\sum_{i=1}^{n}(pred_i - y_i)^2 \qquad (3)$$

Machine learning is considered as an application as well as subset of artificial intelligence. Machine learning provides systems the ability to learn from the experiences and improve without being explicitly programmed. It is a study of algorithms and statistical models that is used by the systems to perform a particular task without being programmed explicitly. These algorithms build models based on sample data called as training data to make predictions to perform the task.

3.1. Linear regression

Linear regression is a supervised learning based machine learning algorithm. It performs regression task. It models the target prediction variable based on input variables. It is used for finding relationship between variables and for the purpose of forecasting. Linear regression predicts the dependant variable (output variable, y) for a given independent variable (input variable, x). It tries to find a linear relationship between input and output. Hence it is coined as linear regression.
Hypothesis function of linear regression is given as

$$y = \theta_1 + \theta_2 x \qquad (4)$$

x : input training data
y: labels to data

The model tries to get the best regression fit by finding the best values for θ_1 and θ_2. The best value for θ_1 and θ_2 can be found by minimizing the cost function.

3.2 *Gradient descent based linear regression*.

Gradient descent is an optimization algorithm in minimizing the cost function in machine learning problems. It is a first order optimization algorithm which means that it takes the first order derivative for updating the parameters. Optimization is required in order to satisfy one of the following needs:

1. To find the global minimum of the objective function. This is possible in the case of convex cost function i.e. any local minimum is a global minimum.
2. To find the minimum value of the cost function in its neighbourhood in case of a non-convex cost function.

In each iteration of gradient descent algorithm, the parameters are updated in the opposite direction of gradient of the objective or cost function where the gradient gives the direction of steepest ascent.

Learning rate

The size of steps we take is called as learning rate. With high learning rate, more ground can be covered on each step but the risk of overshooting the lowest point is more since the slope is constantly changing. A lower learning rate ensures to move in the direction of negative gradient but the time consumption is more. Hence the choice of learning rate plays an important role.

Steps in gradient descent

1. Let us assume that we have two parameters in the model. Let them be denoted by w and b and be denoted by random numbers.
2. Select a value for learning rate (α). Learning rate determines the selection of step size upon each iteration.
   - If α is small, it would take a long time to converge.
   - If α is small, it fails to converge and result in overshoot
3. If the data are on different scales, then it should be scaled. If the data are not scaled it would result in narrow and tall contour which may lead to slow convergence.
4. On each iteration, the partial derivate of the cost function is taken with respect to each parameter.

$$\frac{\partial J(w)}{\partial w} = \nabla_w J \qquad (5)$$

$$\frac{\partial J(b)}{\partial w} = \nabla_b J \qquad (6)$$

5. The update equations are given as

$$w = w - \propto \nabla_w J \qquad (7)$$

$$b = b - \propto \nabla_b J \qquad (8)$$

6. The above process is reiterated until the error curve becomes minimal and doesn't change i.e. on the convergence of the cost function.
7. Under each iteration, the descent would be in the direction that gives the maximum change as it is perpendicular to level curves at each step.

3.3. Neural network method

Neural network, a computational learning system that uses network of functions to know and translate an input from one form to the specified output. The concept of neural networks was inspired from human brain, the way the neurons of the brain function together to understand the sensory input and convert them into actions. It is one of the many approaches in machine learning algorithm. Neural networks can be used in many real life problems such as image and speech recognition, email spam filtering, medical diagnosis etc.

3.3.1 Important terminologies in neural network

Neuron

Neuron is the basic unit of neural network. It receives the input information, processes the information and generates the output which is either sent to the other neuron or is obtained as the final output.

Weights

When an input is given to the neural network, it is usually multiplied by its weights. Random weights are initialized which is then updated during the training phase. After the training, the data with higher importance is assigned with higher weights and the one with less importance is assigned with lower weights.

Bias

In addition to the weights, another linear component is applied to the input of the neural network. It is called as the bias. It is added to the weight multiplication result of the input. The bias is basically added to change the range of the weight multiplied input.

Activation function

A non-linear activation function is assigned to the linear combination of inputs and weights. The activation function translated the input signal to the output signal. The output of the particular neuron may be represented as

$$f(x_1 * w_1 + w_0)$$

Where

x_1 is the input
w_1 is the weight
w_0 is the bias term
f() is the activation function

Input / Output / Hidden Layer

Input layer is the first layer and the one which receives the input from the environment. The output layer is the last layer of neural network which generates the output. The processing is done in the hidden layers. The manipulation of the incoming data is done at the hidden layers and the output of the layer is passed to the next layer.

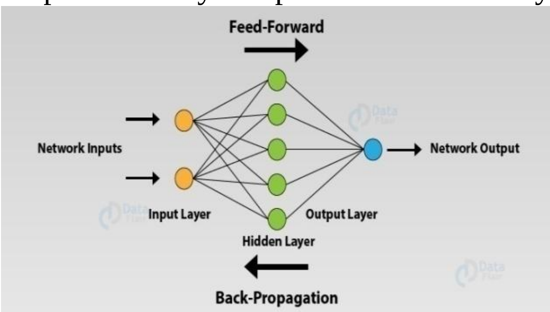

Fig 3Architecture of neural network

3.3.2 Working of neural network

1. The neural network receives input from the external world. The inputs are denoted by x(n).
2. Each input is then multiplied with the corresponding weights. These weights typically represent the strength of interconnections among the neurons in the neural network.
3. The weighted inputs are summed up in the computational unit.
4. A bias term is then added. This is usually done for two purposes. In case the weighted input reaches zero, in order to make the output of neuron non zero, a bias term is added. It can also be added to scale up the system's response.
5. The sum of weighted inputs is now passed through the activation function. The activation function is usually a set of transfer functions used to get a desired output. Some of the commonly used activation functions are binary, sigmoid, tan hyperbolic sigmoid.
6. The output of the layer is passed through successive layers.
7. The actual output of the neural network trained model is obtained from the output layer y(n).

International Journal of Aquatic Science
ISSN: 2008-8019
Vol 12, Issue 02, 2021

3.3.3 Levenberg Marquardt Algorithm

Neural network is trained using Levenberg- Marquadrt algorithm. It is also known as damped least squares. It is used to solve non-linear least squares problem. This problem may arise mainly due to least squares curve fitting.

The algorithm is given as below.

1. Start with randomized update of parameter by rule.
2. Estimate the error of the parameter.
3. If the error has increased as a result of update, the weights are reset and amplify λ by a factor of 10. Repeat the first step and estimate new error value.
4. If the error has decreased, keep the weights at new value and reduce λ by a factor of 10.

2. RESULTS AND DISCUSSION

4.1 Linear regression

The linear regression method was applied on the input-output data set. Various transfer function models were obtained for the choice of number of poles and zeros. They were compared for their performance indices. The model for the selection of 3 poles and 2 zeros were found to provide better results. The obtained parameters for the choice of poles and zeros was

$$\theta = \begin{pmatrix} 0.0140 \\ 0.0012 \\ -1.6524 \\ 0.6948 \\ -0.0001 \end{pmatrix}$$

The output of the system corresponding to the above parameters is shown in Fig 4.

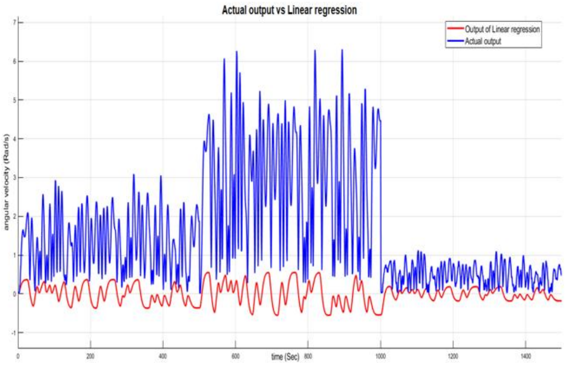

Fig 4 Output of model obtained through linear regression.

The comparison of various models for the choice of poles and zeros are given in the Table 2.

Table 2 Comparison of various models in linear regression

No. of Zeros	**No. of poles**	**Mean square error**	**Integral square error**
1	2	$6.19*10^{19}$	$4.8*10^{32}$
2	2	4.6052	6917.6
2	3	4.6071	6910.4
3	3	5.5590	7028.9
3	4	6.6743	7259.6

But, the integral square error and mean square error is found to be too large. Hence Gradient descent algorithm is implemented and the performance indices are noted.

4.2 . *Gradient descent based regression*

Here, the tuning parameter is the adaptation gain or the learning rate α. The model with the adaptation gain of 0.04 is found to have better performance characteristics. The parameters obtained are given as follows.

$$\theta = \begin{pmatrix} -0.0564 \\ -0.0670 \\ -0.0375 \\ 0.0407 \end{pmatrix}$$

The output of the model for the choice of adaptation gain of 0.04 is shown in Fig 5.

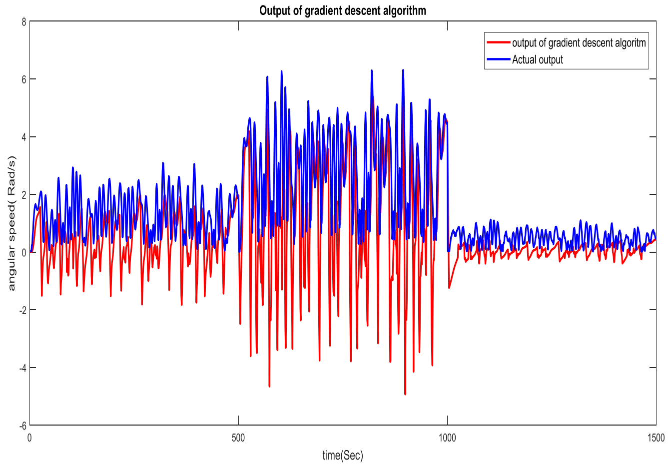

Fig 5 Output of the model obtained through gradient descent algorithm

The output of the model for various values of the adaptation gain are observed and compared based on their performance indices which are shown in Table 3.

Table 3 Comparison of various models in gradient descent

Adaptation gain	**Mean square error**	**Integral square error**
0.5	INFINITY	INFINITY
0.1	4.5201	6780.2
0.05	3.4087	5113.8
0.04	3.3810	5071.5
0.03	3.4327	5149.1

0.01	3.9739	5960.8

The ISE and MSE are still found to be very high and gradient descent algorithm suffers from various convergence issues. Hence identification based on neural networks is done.

4.3 Neural networks algorithm

Neural network was trained for different choice of delayed inputs and number of neurons. The choice of delayed inputs 9 and number of neurons to be 10 was found to have better performance characteristics. The activation function used here is sigmoid activation function. The obtained weights are

Input layer weights:
$[0.35 \quad 0.25 \quad 2.05 \quad -0.68 \quad -0.05 \quad -0.05 \quad -1.65 \quad 0.24 \quad -1.06 \quad 0.07]$

Hidden layer weights: $[-0.23 \quad 0.82 \quad 0.06 \quad -0.08 \quad 0.15 \quad -0.44 \quad 0.07 \quad 0.05 \quad 0.005 \quad -0.06]$

Bias:
$[0.25 \quad -0.55 \quad 0.34 \quad -2.07 \quad -0.17 \quad 0.34 \quad -2.50 \quad 1.53 \quad -1.43 \quad -1.19]$

It is noted that the models developed using neural network algorithm exhibits better performance characteristics compared with the ones obtained through regression and gradient descent algorithm.

The output of the model trained through neural network algorithm is given in Fig 6.

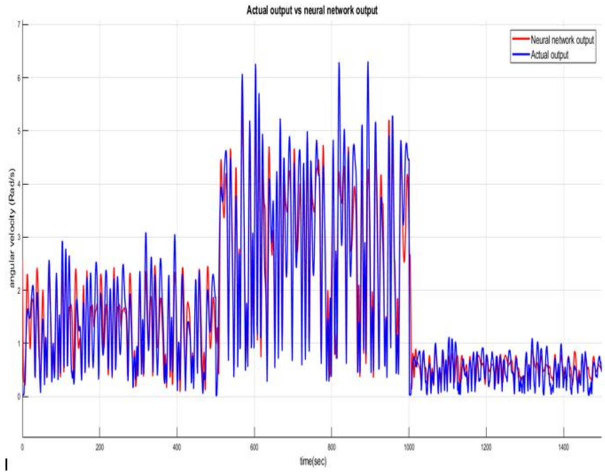

Fig 6 Output of neural network trained model

The performances of various models are given in Table 4.

Table 4 Comparison of various neural network trained models

Hidden neurons	**Delay**	**Mean square error**	**Integral square error**
9	5	2.1155	3167.7
9	2	1.7203	2573.6
10	2	1.6344	2446.9

10	5	1.3501	2020.3
11	2	1.4293	2140.6
11	5	21.6132	3239.0
10	9	0.4055	604.2

3. CONCLUSION

From the results, it could be concluded that linear regression method could not capture the non-linearity in a non-linear system. It works well only for linear systems. Only local minimum could be achieved in case of linear regression method. Gradient descent method provided better results when compared with linear regression method. This method was able to capture the local minimum as well as global minimum. But still, the integral square error and mean square error was found to be high.

The neural network based model with input delay to be 9 and hidden neurons to be 10 with sigmoid activation function tend to obtain better result. It was found that the non-linearity in the system was captured to a better extent and the mean square error and integral square error were also proved to be comparatively lesser.

4. REFERENCES

[1] Ghauri S A, Sohail M, "*System identification LMS, NLMS and RLS*", IEEE Student Research and Development ,pp 65-69,2013
[2] Ljung.L, "*Some aspects on nonlinear system identification*", IFAC symposium on system identification, 2006, vol 39, pp. 553-564.
[3] Long C, Weichuan L, Zeng-G H, "*Neural network based non-linear model predictive control for Piezo electric actuator*", IEEE transactions on industrial electronics, 2015,vol 62, No 12, pp. 7717-7727.
[4] Nidhil, K. J., Sreeraj, S., Vijay, B, "*System identification using artificial neuralnetwork*", International conference on circuit power and computing technologies, 2015.
[5] Schoukens M, Wigren T and, "*Coupled Electric Drives Data Set and Reference Models",* Technical Report, Department of Information Technology, Uppsala University, 2017.
[6] Stodard J G, Welsh J S, "*Data driven modeling of coupled electric drive system using regularised basis function volterra kernels*", International Conference on intelligent robotics and applications, proceedings part I, pp 475-485, 2018.
[7] Tarek A. Tutunji,"*Parametric system identification using neural networks",* Applied soft computing, *2016* ,vol 47, pp. 251-261.
[8] Zheng W X, "*On a least squares based algorithm for stochastic linear systems*", IEEE transactions on signal processing, 2000,vol 46, no. 6, pp 1631-1638.