To Identify a Second Order Plant Approximating a Neural Network using Neural Network Based Predictive Control Technique

Priyaranjan Mandal#1, Binay Biswas*2

#1Associate Professor, Department of Applied Physics
92, A P C Road, Kolkata-700009, India
*2Technical Superintendent, Department of Applied Physics
92, A P C Road, Kolkata-700009, India

Abstract — In this paper, a second order plant is considered for identification using NN-predictive control technique. The neural network based predictive controller is configured based on MATLAB 7.0. This neural network controller uses a neural network model of plant. It predicts the future performance of the actual plant. The controller uses to calculate the control input. The control input will optimize the performance of the plant. It optimizes the performances over a specified future time horizon.

At first, the procedure of model predictive control takes identifies the neural network plant model. Then the predictive controller uses the plant model and predict the future performance of the model. Thus the term ‘predictive’ signifies to identify the model plant.

We have considered an arbitrary second order plant to observe the performance of the controller how it can predict the future performance of the model plant. The result obtained via this procedure is very close approximation to the actual behaviour of the plant.

Keywords — Training a neural network, plant identification, model predictive controller.

I. INTRODUCTION

The method of neural network (NN) based model predictive control technique is used now-a-days to identify [2,5,13, 14] both linear and non-linear systems. This method is used to analyze the problems related to power systems, electrical machines, control systems and different process plant industries. [6, 7, 9, 11, 12]

This paper considers an arbitrarily chosen typical second order system. In the first stage, model predictive controller [4] configures a neural network in order to represent the so called ‘forward dynamics’ of the actual plant. This is configured by learning the behavior of the actual second order plant through training [1,6] the network. The 'process of learning' uses a algorithm or method of training (trainlm) to train [1,7] the network for a meaningful identification. A prediction error is calculated. This is the error which is calculated between the actual plant output and trained (trainlm) neural network model plant output. This error is called prediction error as it is produced by the prediction of the NN predictive controller. This predictive error signal is used as the training signal of the neural network going to be trained. Ultimately, the neural network configured via this procedure of learning followed by training (trainlm) is the representative of the actual second order plant considered for identification [3, 5]. The neural network model plant uses two information to predict future behaviors of the actual plant output over a specified time horizon. These two information categorically are —previous inputs and previous plant outputs.

The main objective of the work is as follows:
(i) To realize the simulation diagram of the proposal.
(ii) To design the model predictive controller.
(iii) The develop the neural network plant model.

In section 2, the description of the model predictive controller is furnished. In section 3, mathematical model of the actual plant has been shown. In section 4, different types of neural network controllers used for the purpose of identification have been described. In section 5, the module of NN- predictive or predictive neural network controller (PNNC).is shown. In section 6, the simulink diagram of the proposed scheme has been shown. In section 7, the design of model predictive controller is considered. In section 8, design of neural network (NN) plant model has been considered. In section 9, the review of the result has been furnished. In section 10, the conclusion of this work has been discussed.
II. THE DESCRIPTION OF MODEL PREDICTIVE CONTROLLER

The method of model predictive control is solved using receding horizon technique. The predictive control technique predicts the plant response over a specified time horizon. The predictions are used by a numerical optimization program to determine the control signal that minimizes a performance criterion over a specified time horizon. This is described as follows:

\[
J = \sum_{j=N_1}^{N_2} (y_r(t+j) - y_m(t+j))^2 + \rho \sum_{j=1}^{N_u} (\hat{u}(t+j-1) - \hat{u}(t+j-2))^2
\]

(1)

where, \(N_1\), \(N_2\) and \(N_u\) describes the horizons over which the control increments and tracking error are estimated. The term \(u'\) is a variable and it is the tentative control signal. The symbol \(y_r\) is assumed as the desired response. The symbol \(y_m\) is considered as the network model response. The symbol \(\rho\) is considered as it determines the contribution that the sum of the squares, as shown in the above equation, of the control increments gives on the ‘performance index’ \(J\).

The model predictive controller consists of two parts- the neural network model plant and a block called optimization block. The function of optimization block is to determine the values of \(u'\) which, on the other hand, minimizes the performance index, \(J\). The optimal value of \(u'\), for which \(J\) is minimized, is inputed to the plant.

III. THE MATHEMATICAL MODEL OF THE ACTUAL PLANT

The mathematical model of the problem, as is considered as an arbitrary one, has been developed in state-space domain. The state space representation of same has been shown below.

\[
\begin{bmatrix}
\dot{x}_1 \\
\dot{x}_2
\end{bmatrix} = \begin{bmatrix}
0 & 1 \\
-25 & -5
\end{bmatrix} \begin{bmatrix}
x_1 \\
x_2
\end{bmatrix} + \begin{bmatrix}
0 \\
1
\end{bmatrix} u
\]

(2)

\[
y = \begin{bmatrix}
25 \\
0
\end{bmatrix} \begin{bmatrix}
x_1 \\
x_2
\end{bmatrix}
\]

(3)

where, \(x_1\) and \(x_2\) are state vectors. The output of the system is designated by \(y\). Here, \(u\) is considered to be a scalar and used as a control input. The response of the actual plant (as per eq. no. 2 & 3) is shown in Fig. 1.

Fig. 1. Behavior of the actual plant. (ref. eqn. 2 & 3)

The simulink model has been developed based on state space model for the actual plant in the MATLAB.

IV. DIFFERENT TYPES OF NEURAL NETWORK CONTROLLERS [13,14]

Neural Network Toolbox in MATLAB 7.2 offers three well known methods for identification of different linear and non-linear systems of power systems, machines and control systems. [13,14]

The methods use three controller as - (i) Narma-L2 Controller, (ii) NN Model Reference Controller and (iii) NN predictive or predictive neural network controller (ref. Fig.2). In this paper, we take the opportunity to utilize of NN predictive or predictive neural network controller (PNNC).

V. THE MODULE OF NN PREDICTIVE OR PREDICTIVE NEURAL NETWORK CONTROLLER (PNNC)

The module or block diagram of the NN predictive model controller or predictive neural network controller (PNNC), as available in Neural Network Toolbox in MATLAB 7.2, is shown in Fig. 2. [13,14]

![Fig. 2: The module of predictive neural network controller (PNNC)]
This module consists of neural network (NN) model plant block and an optimization block. Some performance index has been considered mathematically which will be zero for an optimal training of the NN plant model.

The Fig. 3, shows the detailed view of the module as shown in Fig. 2. This can also be defined as the ‘view under the mask’.

VI. THE SIMULINK DIAGRAM OF THE PROPOSED SCHEME

In Fig. 4, the simulink diagram for the proposed scheme has been furnished. The greenish block is the predictive neural network controller (PNNC). The actual plant as per eq. 2 and 3 has been developed in state space domain. This is shown as subsystem and is connected with the predictive neural network controller (PNNC) in one end. The other end of the subsystem is connected to a plotter ‘X(2Y)’. The plant has been developed in state space domain based on MATLAB simulink 7.2. The subsystem dedicates for state space actual plant model.

VII. DESIGN OF NN MODEL PREDICTIVE CONTROLLER

The neural network Predictive control block is considered to develop/design the model predictive controller or NN Predictive controller(NNPC). The variable $N_1$ is kept fixed at the value equals to numerical one. The variables $N_2$ and $N_u$ are adjusted to modify the horizons of the controller. The weighting parameter $\rho$ may also be updated for minimizing the ‘$J$’. The search parameter $\alpha$ is used to modify the optimization. It determines the successful optimization steps. [10,11]

VII. DESIGN OF NEURAL NETWORK (NN) PLANT MODEL

The neural network plant model predicts future plant outputs. These predictions are utilized by the optimization algorithm to determine the control inputs, $u'$ that optimize the future performance.

Fig. 5: The Neural Network predictive controller block

Fig. 6: Procedure of learning through training (trainlm) of the model neural network plant

The algorithm generates training data by applying a series of random step inputs to the Simulink plant model. The training data defined as potential training data comes out in a way as shown in the Fig. 7. [13,14]
Fig. 7: The potential training data as generated during the process of training of the neural network plant model.

If the algorithm is allowed to accept the data, training (\textit{trainlm}) of the model plant initiates. The process of training (\textit{trainlm}) learns the behavior of the actual plant and trains the neural network model plant accordingly. This is a mode of batch training.

Fig. 8: Training data produced during process of learning through training (\textit{trainlm}) of neural network based plant model.

As soon as the procedure of training completes, the result of response of training the plant model is observed in the Fig.8.

Fig. 9: Validation data produced during process of learning through training (\textit{trainlm}) of neural network based plant model.

Fig. 10: Testing data produced during process of learning through training (\textit{trainlm}) of neural network based plant model.

The Fig. 11 shows the report which is produced during the training (\textit{trainlm}) of the neural network model plant.

Fig. 11: Report of training (\textit{trainlm}) of the neural network based plant model.

The report shows the configuration of the neural network being trained and it is now ready to predict the future behavior of the model NN plant. Two hundred nos. of iterations have been
considered. The prediction error has been calculated based on mean square error (MSE). The performance index, J, after 16 no. of iterations is found to be $6.21 \times 10^{-17}$. The value of J becomes minimum and reduces to zero after 200 nos. of iterations.

**IX. THE REVIEW OF THE RESULT**

The NN-predictive controller has predicted the dynamic behavior of the actual plant. As per these predictions the neural network plant model has been configured and trained accordingly. This NN-plant model is representing for the actual plant. Thus the actual plant is identified. The green colored curve is the output of the neural network based plant model. The curve shown in blue color is the output of the actual plant as shown in Fig.1.

![Identification of the plant](image)

**Fig. 12:** Comparison of outputs of NN model plant (green) and the actual plant (blue)

It is seen that output of the NN model plant (identified plant) goes very close approximation to the actual one.

**X. CONCLUSION**

A simple linear second order system has been considered for identification via neural network (NN) predictive control technique. The NN predictive control technique predicts about the behavior of the actual plant over a specified time horizon. At first, the predictive controller configures an elementary neural network. This is configured by learning the behavior of the second order plant model through training the network. The ‘process of learning’ uses a algorithm or method of training (trainlm) to train the network. Two hundred nos. of iterations have been considered. After this, training is completed and the value of performance index, as shown in eq. no. 1, is found to be zero.

In Fig. 12, the output of the neural network plant model, the representative of the actual plant, has been shown. The behavior of the actual plant has also been shown for comparison. The response of the neural network based plant goes very close approximation to the actual plant.

**REFERENCES**


