Measurement 63 (2015) 152–158

Contents lists available at ScienceDirect

Measurement

journal homepage: www.elsevier.com/locate/measurement

A comparison of artificial neural network and extended Kalman filter based sensorless speed estimation

Zafer Aydogmus [a,*], Omur Aydogmus [b]

[a] Fırat University Technology Faculty, Electric and Electronics Engineering, Elazığ, Turkey
[b] Fırat University Technology Faculty, Mechatronics Engineering, Elazığ, Turkey

ARTICLE INFO

Article history:
Received 11 August 2014
Received in revised form 3 October 2014
Accepted 2 December 2014
Available online 12 December 2014

Keywords:
Speed estimation
Neural networks
Sensorless control
Model-based estimation

ABSTRACT

In industry speed estimation is one of the most important issue for monitoring and controlling systems. These kind of processes require costly measurement equipment. This issue can be eliminated by designing a sensorless system. In this paper we present a sensorless algorithm to estimate shaft speed of a dc motor for closed-loop control using an Artificial Neural Network (ANN). The method is based on the use of ANN to obtain a convenient correction for improving the calculated model speed. Three architectures of ANNs are developed and performance evaluations of the networks are performed by three performance criteria. After the evaluations, Levenberg–Marquardt backpropagation algorithm is chosen as learning algorithm due to its good performance. The speed estimation performance of developed ANN was compared with Extended Kalman Filter (EKF) under the same conditions. The results indicates that the proposed ANN shows better performance than the EKF. And ANN model can be used for speed estimation with reasonable accuracy.

1. Introduction

In industry, electrical motors, hydraulic and pneumatic systems are used to perform all of the required mechanical movements. The motors are needed to drive these hydraulic and pneumatic systems. Thus, electric motors play an important role in industrial applications. Most of these motors need speed control to implement some of the tasks. To obtain a closed-loop control system, some parameters such as current, speed and rotor position need to be known. The measurements of all these mentioned variables increase the total cost of the system. The shaft-position encoders and tacho generators are used to measure the speed/position information of motor. Shaft mounted speed/position measurement causes an increase in the cost and volume of the motor drive system. In the hard work conditions affect the robustness and reliability of drive system negatively [1–3]. The estimation of motor speed without using any kind of measurement devices such as encoder or a tacho generator eliminates these mentioned problems.

The observers can be used to estimate the position/speed of the rotor. The observers are more secure tools than actual sensors because of not using any kind of shaft speed sensors [4–6]. However, an observer needs some mathematical equations and some measurements of motor such as current and voltage. In closed-loop motor drive systems, the level of estimation accuracy of speed information has an important role. In addition, the information of speed, acceleration and jerk are required for some industrial process [7,8].

The artificial neural network has been known as powerful tool capable of estimating for linear and non-linear systems. ANNs can be used to simulate a nonlinear dynamic systems with any desired degree of accuracy [9]. Various ANNs have been proposed for speed and position estimation and control of electrical machines [10,11]. In literature,

* Corresponding author. Tel.: +90 424 237 0000.
E-mail address: zaydogmus@firat.edu.tr (Z. Aydogmus).

http://dx.doi.org/10.1016/j.measurement.2014.12.010
0263-2241/

several methods such as sliding-mode [12], Kalman filter [13], extended Kalman filter [1,14], and the particle filter [15–17] have been used to estimate the state variables of a nonlinear sensorless system.

In this study, a neural network based speed estimation for dc motors was presented. A sensorless speed control system was developed to compare the ANN and EKF by using MATLAB simulation program. The simulation model consists of two parts: digital signal processor model (DSPM) and a virtual real-time system (VRTS). The ANN and EKF were performed simultaneously and at the same conditions to compare their performance of speed estimations. The results of motor performance are presented at various speed references in detail. When the comparison of the performances were performed, it was seen that the ANN had very good estimation results for speeds of a dc motor than EKF for different speed references.

The organization of paper is as follows: Section 2 outlines overview of mathematical model of the system in detail. In Section 3, EKF and proposed ANN were introduced. In Section 4, details of the system structure and simulation results for the performance of the ANN and EKF are given. And simulation results are reported and discussed. Then, the conclusion is presented in Section 5.

2. Mathematical model of the system

The machine model consists of two main mathematical equations. One of them is an electrical and other is a mechanical equation. The details of expressed equations are given in [1]. In these equations motor current i_a, motor shaft speed ω are two state-variables. The mechanical, electrical and load equations of the motor are given together in matrix form as follows:

$$\frac{d}{dt}\begin{bmatrix}\omega\\ i_a\end{bmatrix} = \begin{bmatrix}-B_m/(mL^2+J) & k_e/(mL^2+J)\\ -k_e/L_a & -R_a/L_a\end{bmatrix}\begin{bmatrix}\omega\\ i_a\end{bmatrix} + \begin{bmatrix}(-mgL\cdot\cos\theta - T_f)/(mL^2+J) & 0\\ 0 & 1/L_a\end{bmatrix}\begin{bmatrix}1\\ u_a\end{bmatrix} \tag{1}$$

where R_a is the resistance and L_a is the inductance of the armature winding. u_a is the armature supply voltage, J is the inertia, B_m is viscous friction coefficient, T_f is the Coulomb friction torque. Also the mass of the load is m, the length of the arm is L and the gravity is g. Estimated angular velocity value ω is used to calculate the angular acceleration term θ. The state-space model of dc motor can be obtained by using the voltage equations. The load model of motor is given in Fig. 1.

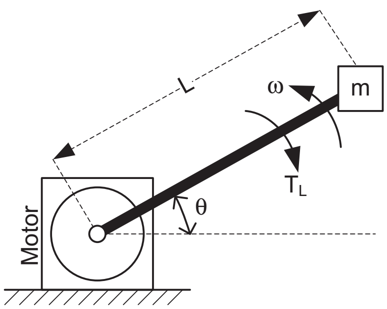

Fig. 1. Nonlinear load model for dc motor.

The speed of motor was estimated by both ANN and EKF using motor model with Eq. (1). First step; motor model block calculates the motor speed and current. In second step ANN and EKF estimate the motor speed as mentioned in following section. Simulink blocks of load model and motor model is shown in Fig. 2.

3. Estimation algorithms

3.1. ANN algorithm

ANN is a powerful tool that is able to represent nonlinear systems and applicable to a wide variety of fields. Neural network consists of connected neurons that are the basic information processing structures. They communicate by sending signals to each other along weighted connections. The learning rule modifies connection weights during the training process. Learning process is a mathematical method which improves the neural network's performance by giving weights and biases computed from a set of training data. After the learning process the testing data are used to check the validation of net. The initial weights and biases are usually assigned randomly. Activation function is used to calculate the output of neurons. For each neuron in the hidden layer the output y_j is given by

$$y_j = f\left(\sum w_{ji}x_i\right) \tag{2}$$

where x_i is input signals, w_{ji} is connection weights and f is activation function. A sigmoidal activation function is used to calculate the output of neurons. The sum of squared differences between the desired and actual values of the output neurons E is given by

$$E = \frac{1}{2}\sum_j (y_{dj} - y_j)^2 \tag{3}$$

where y_{dj} is the desired value of output neuron j and y_j is the actual output of that neuron [18,19].

In this work, the input–output data have been normalized before the initiation of the training of the ANN. In this normalization, the maximum values of the input and output data are determined as follows:

$$x_{i,\max} = \max(x_i(j)) \quad j = 1,\ldots,N_j \quad i = 1,\ldots N_i \tag{4}$$

where N_j is the number of patterns in the training set and N_i is the number of neurons in the input layer. For output layer;

$$y_{k,\max} = \max(y_k(j)) \quad j = 1,\ldots,N_j \quad k = 1,\ldots N_k \tag{5}$$

where N_k is the number of neurons in the output layer. Normalized by these maximum values, the input and output variables are given as follows: