

Numerical and neural network modeling and control of an aircraft propeller

Hami Tourajizadeh^a, Soleiman Manteghi^b, *, Saeed Rafee Nekoo^b

^a Mechanical engineering department, faculty of engineering, tehran, Iran

^b Independent Researcher, Tehran, Iran

ARTICLE INFO

Article history:

Received: 19 July 2017

Accepted: 16 October 2017

Keywords:

Aircraft propellers

Modelling and control of DC motor

Training algorithm

Levenberg-Marquardt

ABSTRACT

In this paper, parametric and numerical model of the DC motor, connected to aircraft propellers are extracted. This model is required for controlling trust and velocity of the propellers, and consequently, an aircraft. As a result, both of torque and speed of the propeller can be controlled simultaneously which increases the kinematic and kinetic performance of the aircraft. Parametric model of the motor is derived by conducting standard tests such as locked rotor test and step and sine wave input one. In order to derive a neural network and numerical model, a set of sinusoidal, triangular, and random step signals are applied as the input to the motor and its speed is recorded as an output. Neural network of the motor is extracted by using these datasets and considering a multilayer perceptron (MLP) neural network structure with Levenberg-Marquardt training method. Results of the numerical model and parametric model are compared and validated by experimental implementations. The superiority of the proposed method is also shown respect to traditional PID algorithm.

1. Introduction

Over the past 20 years, advances in propeller-driven propulsion and light-weight materials and electronics were led to current-day RC/UAV fixed-wing aircraft configurations with thrust-to-weight ratios of nearly 2:1. This level of performance combined with large control surfaces and lifting ones immersed in high-speed propeller wash, was produced a “new breed” of ultra-agile aircraft that can be flown over an envelope that ranges from conventional fixed-wing cruise flight to a stop-and-stare hover attitude, and nearly anything aerobatic in the range. This extreme performance presents new challenges in propeller/airframe modelling and simulation [1].

Methods for modelling propeller aerodynamics in cruise flight are well known and literature on this subject abounds, [2]. Challenges exist such as unsteady effects, in modelling the transition from level cruise flight to hover. More generally, propeller aerodynamics at any flight condition, that is, any aircraft attitude and speed, must be addressed to capture full-envelope flight dynamics, which can occur in upsets or ultra-agile/aerobatic maneuvers. Propeller modelling must also include direct propeller force and moments as well as propeller wash effects on downstream surfaces and the opposite scenario. Finally, all of these effects must be combined in a “seamless” and computationally-efficient methodology when applied in a real-

time piloted simulation environment. Overall scheme of the propellers of an aircraft is depicted in Fig. 1.

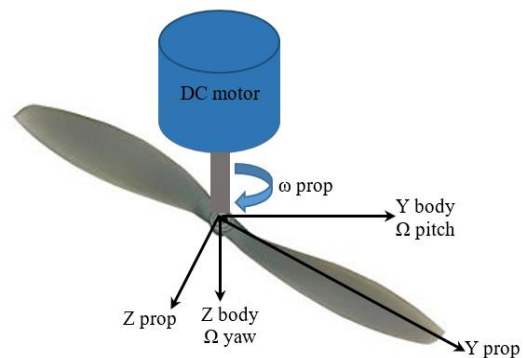


Figure 1. Aircraft propellers.

Since the control approach is model based, not only the model of the propellers' aerodynamics is required, but also the model of the motor should be extracted to achieve more accuracy. Inverse kinematics and dynamics are required for precise controlling of the propellers of the aircraft. Aerodynamics and control of the marine propellers were extracted in previous literature [3]. In order to implement the controlling algorithm, speed-torque chart of the motor needs to be extracted by experimental tests, since the parameters of the motor are not available. The voltage is applied

* Corresponding author. Tel.: +989122362359; Tourajizadeh@khu.ac.ir

to the motor as the input and the output torque is recorded as the motor output, then input-output chart is determined. Any input and output between two points of the chart is provided by interpolation. PWM duty cycle is set, according to the input voltage level; hence, this PWM signal is applied to control the torque of the motors. The electrical model of the motors was needed to implement the mentioned feedback linearization technique to control the rotation of the propellers. Some research projects were performed in this area so far. Urrea [4] estimated the dynamic model of a motor and controlled the motor using an extracted dynamic model and simulated its dynamic behavior based on its estimated dynamic parameters. The results were verified by comparing with simulation data. It can be seen that the exact dynamic model of a motor was required there to control a system. Anandaraju [5] employed a PID controller for DC motors. The gains were tuned by the aid of a proper iterative method. Genetic algorithm is a good choice for optimizing the PID gains for speed control of DC motors. Different objective functions were used for tuning the gains. Ohishi [6] used proportional control action (P), to minimize the position error of DC motors. The unwanted transient and permanent errors which were the result of friction and parametric uncertainties were compensated by employing a new method based on observation theory and microprocessor programming. The lack of these researches was the absence of feed-forward term which prevents us from using them in high dynamic response applications of dynamic systems. Sequential training neural network based on feed-forward (FFNN) was proposed in [7] by Hoque, to control the speed of permanent magnet of DC motors. Artificial neural network (ANN) control is a good solution for the systems with extreme nonlinearity of the dynamic, unexpected parametric uncertainties or huge disturbance. The system improves itself using an auto training model in an online way. This algorithm was implemented on a permanent magnet DC motor. Liu [8] controlled the position of a DC servo motor by the aid of PID in LabVIEW environment. The online control was realized using data acquisition (DAQ) card. Fuzzy controller together with PID (FPID) were used by Altayef [9] to control the position of DC motors in LabVIEW environment. Again, DAQ board was used to control the motor in an online scheme. The experimental results showed that FLC had a good performance. Yan-hong et. al [10] optimized the gains of PID controller of servo motors. The dynamic of the system was used to control by the use of PID. In order to make the system robust, neural network equipped by an observer was added to the controller. Optimizing and tuning the gains of PID was done employing genetic algorithm. The proposed algorithm was implemented for motor driver and its efficiency was verified by experimental tests.

In this paper, electrical model of a DC motor is extracted by numerical and parametric method. These models increase the accuracy of the rotational speed of the propellers. Experimental test is done to estimate the parametric model by the aid of wave form output of the motors. Numerical model of the motor is also proposed here using neural network model of the motor by implementing the numerical analysis of the input and output data of the motor. Two approaches are compared and analyzed to confirm their efficiency and advantages. Neural network modelling of the DC motor and its parameters are obtained. Then the MLP neural network is presented and training by Levenberg-Marquardt algorithms is performed. Finally, in order to confirm the proposed modelling strategy and its efficiency, experimental results are compared and analyzed.

2. Equivalent circuit and modelling of DC motor

Computed torque method is used for precision control of the propellers of an aircraft. Torque and angular velocity of the motors are estimated based on the inverse dynamics using aerodynamics of the propellers. Here, not only the aerodynamics of the propellers are required, but also the model of the motors are needed in this method. The aerodynamics of the propellers can be presented as [1, 3]:

$$\begin{aligned} T &= \rho n^2 D^4 C_T \\ Q &= \rho n^2 D^5 C_Q \end{aligned} \quad (1)$$

where T and Q are thrust and torque of the propellers, ρ is the air density, n and D are the respectively rotational speed and diameter of the propellers and C_T , C_Q are thrust and torque coefficients of the propellers which could be evaluated using the related look up tables. Based on the above formulation of propellers' dynamics, and in order to control the propellers within a desirable rotational speed, the model of the motor is necessary to evaluate the required control voltage. To produce a desirable rotational speed, a specific amount of torque load is exerted on the propellers. So two approaches are proposed here to extract a model of a DC motors.

Electromagnetic torque is calculated in (2). Here i_a is the motor current, e is the electromotive force, ω_m is the angular velocity and K_b is the electromotive constant of the motor [11]:

$$T_e = \frac{e i_a}{\omega_m} = K_b i_a \quad (2)$$

Electrical energy as the input of a DC motor is converted to mechanical energy which is equivalent to air gap torque multiplied by rotational speed of the rotor. So the torque of a DC motor is controlled by armature current.

The load is accelerated by electromechanical torque as (3):

$$J \frac{d\omega_m}{dt} + B_1 \omega_m = T_e - T_l = T_a \quad (3)$$

where J is inertia of rotor, B is the friction constant, T_e is the electrical torque and T_l is the load torque. Therefore, T_a is the torque that accelerates the rotor [11].

State space can equation be represented as (4):

$$\dot{X} = AX + BU \quad (4)$$

where $X = [i_a \ \omega_m]^T$ is state variable and $U = [V \ T_l]^T$ is input vector consist of voltage (V) and load torque (Tl), A and B are as follows:

$$A = \begin{bmatrix} -\frac{R_a}{L_a} & -\frac{K_b}{L_a} \\ \frac{K_b}{J} & -\frac{B_1}{J} \end{bmatrix}, \quad B = \begin{bmatrix} \frac{1}{L_a} & 0 \\ 0 & -\frac{1}{J} \end{bmatrix}$$

where R_a and L_a are resistance and inductance of the armature coil respectively. The roots of the system according to A matrix are (5):

$$\begin{aligned} \lambda_1, \lambda_2 &= \frac{-\left(\frac{R_a}{L_a} + \frac{B_1}{J}\right) \pm \sqrt{\left(\frac{R_a}{L_a} + \frac{B_1}{J}\right)^2 + 4\left(\frac{R_a B_1}{J L_a} + \frac{K_b^2}{J L_a}\right)}}{2} \end{aligned} \quad (5)$$

Always right value of A is negative real value. So DC motor is stable in open loop applications. Block diagram of DC motor is shown in Fig. 2 and according to that, angular velocity with respect

to the motor voltage and angular velocity with respect to torque is extracted and represented in (6) and (7) respectively:

$$\frac{\omega_m(s)}{V(s)} = \frac{K_b}{s^2(JL_a) + s(B_1L_a) + (B_1R_a + K_b^2)} \quad (6)$$

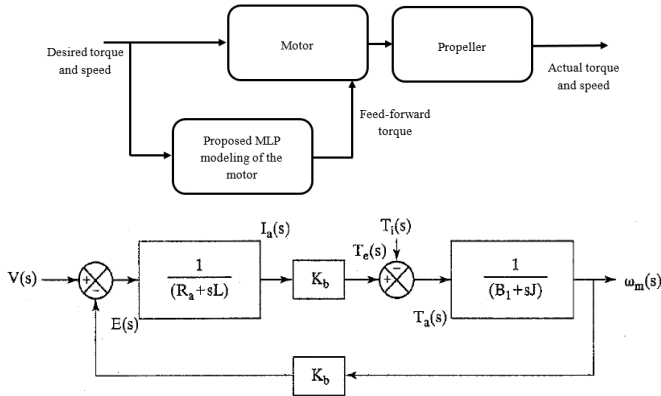


Figure 2. Block diagram of DC motor [11].

Therefore, the angular velocity of a DC motor is controlled by its input voltage and load torque. A proportional-integral-derivative (PID) controller is employed in this paper, which is a control loop feedback mechanism widely used in industrial control systems and a variety of other applications requiring continuously modulated control. The general control law can be expressed mathematically as $u(t) = K_p e(t) + K_i \int_0^t e(\tau) d\tau + K_d \frac{de(t)}{dt}$, where K_p is proportional gain, K_i is integral gain and K_d is derivative gain, all of them non-negative.

3. Neural network algorithm

The Levenberg-Marquardt algorithms is used for NN (neural network) leaning. A review of these algorithms is represented here [12]:

All of the NN inputs are applied to the network and all of these outputs are calculated. Afterward, Sum of squared errors for all of the inputs is computed.

Jacobian matrix for this network is computed by (8)

$$J(x) = \begin{bmatrix} \frac{\partial e_1(x)}{\partial x_1} & \dots & \frac{\partial e_1(x)}{\partial x_n} \\ \vdots & \ddots & \vdots \\ \frac{\partial e_n(x)}{\partial x_1} & \dots & \frac{\partial e_n(x)}{\partial x_n} \end{bmatrix} \quad (8)$$

where $e(x)$ is the NN error for x-th parameters. Variation of NN weights are calculated by (9)

$$\Delta x = [J^T(x)J(x) + \mu I]^{-1} J^T(x)e(x) \quad (9)$$

where μ is training coefficient and I is a unit matrix.

The sum of squares of the errors is recomputed using $x + \Delta x$. If this new summation value is smaller than that computed in step 1, then μ should be reduced by p (usually $p=10$). Then, let $x = x + \Delta x$ and go back to step 1. Otherwise, if the summation value is not smaller than that computed in step 1, then μ should be increased by p and then go back to step 3.

The algorithm is assumed to be converged when the norm of the gradient is less than some predetermined value, or when the sum of squares is reduced to an acceptable level.

$$\frac{\omega_m(s)}{T_l(s)} = \frac{-(R_a + sL_a)}{s^2(JL_a) + s(B_1L_a) + (B_1R_a + K_b^2)} \quad (7)$$

4. Parametric modeling of DC motors

In this section parametric model of a DC motor is extracted by experimental test. An application was implemented in C# for generating desired input to apply to the motors and to record the output of the DC motor simultaneously. An electronic interface circuit was designed and constructed to send the input data from PC to the motor and to receive the motor speed as the output to a PC by RS232 port. The input voltage of the motor is converted to Pulse Width Modulation (PWM) by microcontroller (ATMEGA64). Then it is amplified (L293) and is applied to the terminal of the motor armature. The motor speed is estimated using incremental rotary encoder. The output of the encoders is recorded by microcontroller and is sent to a PC. A scheme of this application and its related electronic circuit is shown in Fig. 3 and Fig. 4 respectively.

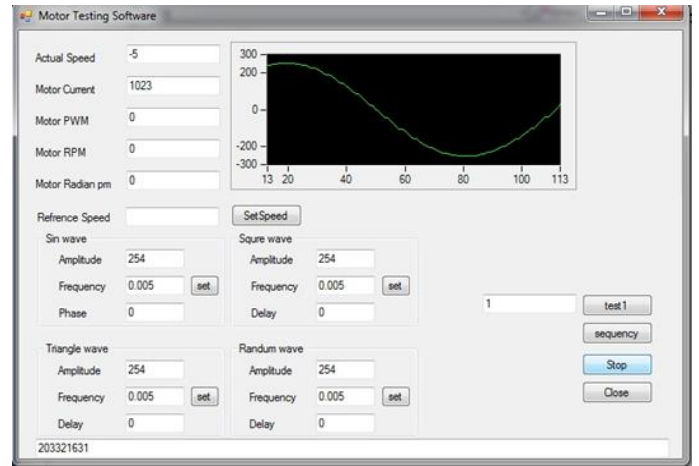


Figure 3. C# application for motor drive.



Figure 4. Electrical board interface.

4.1. Armature resistance

DC component of the armature resistance is computed by applying DC voltage to the motor and locking the rotor. Then the armature currents are measured. The nominal currents of the motor should be considered. DC component of the armature resistance is calculated by (10):

$$R_{dc} = V_{dc} / I_{dc} \quad (10)$$

The experimental result is shown in Fig. 5.

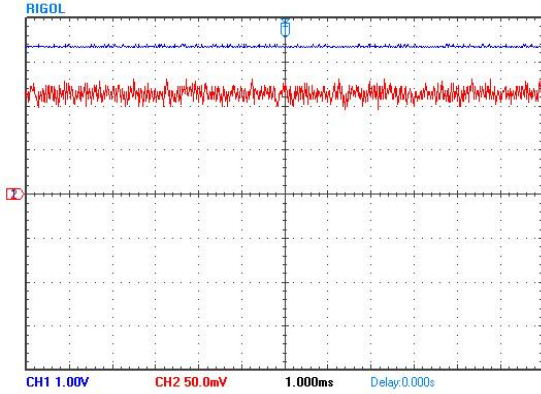


Figure 5. Input voltage and Current of the motor.

The parameters of each waves form can be seen in Table 1, 2 and the related calculated DC component of the armature resistance can be seen in Table 3.

Table 1. Voltage.

parameter	value
Vpp	120mV
Vrms	3.34V

Table 2. Current* R(1.5 ohm).

parameter	value
Vpp	36.0mV
Vrms	124mV

Table 3. Computed resistance (rms stands for root mean square).

buehler	R=1.5			
CH2	Vrms	0.124	I(A)	0.082667
CH1	Vrms	3.34	V(V)	3.34
			R(ohm)	38.90323

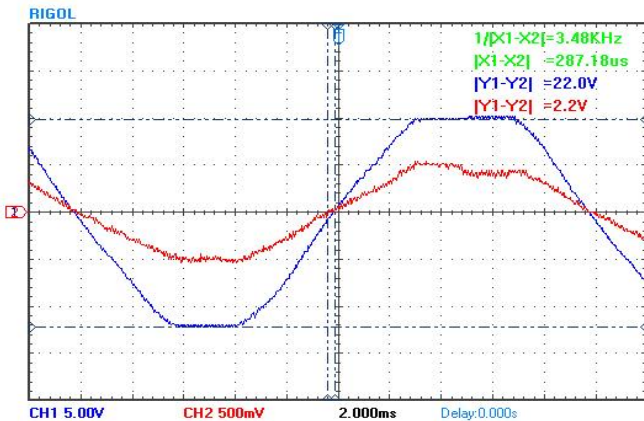


Figure 6. Input voltage and current of the motor.

Table 4. Computed motor inductance (avg stands for average, amp stands for amplitude, rms stands for root mean square).

Current* R	
Vpp	1.92V
Vmax	960mV
Vmin	-940mV
Vavg	17.1mV
Vamp	1.67V
Vtop	755mV
Vbase	-921mV
Vrms	604mV

Voltage	
Vpp	23.0V
Vmax	10.4V
Vmin	-11.6V
Vavg	-722mV
Vamp	20.3V
Vtop	8.82V
Vbase	-11.4V
Vrms	7.24V

Table 5. Computed variance phase of current and voltage.

us	f	w	phi(rad)	phi(degree)
244	50	314.159	0.0766	4.394

Table 6. Computed motor inductance.

tan(phi)=LW/R		L (inductance)		
tan(phi)	R	W	(H)	L(mH)
0.076805355	38.903	314.159	0.00951	9.51

4.2. Armature Inductance

Inductance of the motor is computed by applying a sinusoidal voltage as the input of the motor and recording the motor current as its output. Input/output chart is plotted, and then the variance phase between the input and output of the motor is computed. The rotor is locked during the test procedure, in order to disable the electromotive force (emf) effect on this test. The Inductance of the armature can be computed by (11) [11]:

$$L_a = \frac{\sqrt{V_a^2 - R_a^2}}{2\pi f_s} \quad (11)$$

where f_s (Hz) is the input voltage frequency, R_a (ohm) is the armature resistance, V_a (V) is rms of input voltage, I_a (A) is the RMS of the motor current and L_a is the armature inductance (H). Input and output wave form is shown in Fig. 6. Characteristics of the input and output wave are presented in Table 4. The rotor inductance is computed and is shown in Table 6.

4.3. EMF constant

The shaft is rotated by a prime mover (another DC motor) up to the nominal speed of the motor. The armature coil is open and the voltage induced in this terminal is recorded for any motor speed. The result is plotted in the chart in Fig. 7. The slope of this curve at any specific speed is represented as the emf constant of the motor (V.s/rad). According to Fig. 7 emf constant is computed about 0.238.

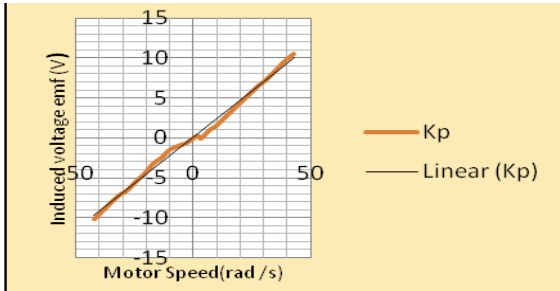


Figure 7. Induced voltage via motor speed.

Table 7. Parameters of the motor together with the propeller (emf stands for electromotive force).

Parameters	value
Momentum of inertia J	0.01 $\frac{\text{kg} \cdot \text{m}^2}{\text{s}^2}$
Mechanical B ₁ damping coefficient	0.1 Nm.s
Emf constant K _b	0.238 $\frac{\text{N} \cdot \text{m}}{\text{Amp}}$
Electrical resistance	38.90ohm
Electrical Inductance	0.01H

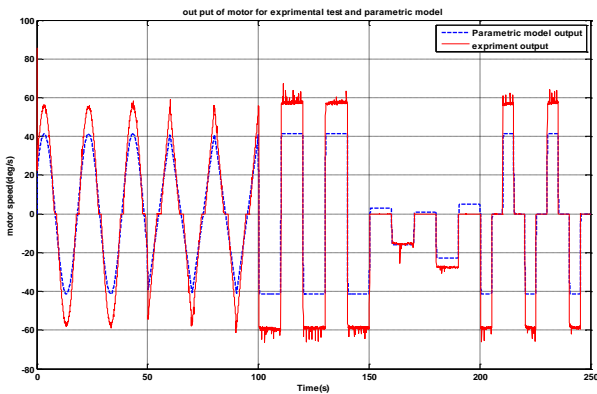


Figure 8. Parametric model output compared to experimental output.

The desired position and velocity of the aircraft can be achieved by controlling the motor speed calculated from inverse dynamics of the propellers dynamics and using the proposed parametric modelling of the motor together with the employed propeller which is shown in Table 7. Parametric model of the DC motor and propeller according to Table 7 was implemented and the input data set were applied to this model. The output of the parametric model was compared with experimental test in Fig. 8. It can be seen that the open loop output of the DC motor with parametric modelling is the same as the output of experimental tests with the same input, but the output amplitude of the profile related to parametric modelling is 15 deg/s less than the output of DC motor. So it can be concluded that this model of DC motor is not quietly precise and should be improved using some numerical strategies.

4.4. MLP neural network model of DC motors

A multilayer perceptron is a class of feed-forward artificial neural network, consists of at least three layers of nodes [13]. Except for the input nodes, each node is a neuron that uses a nonlinear activation function and utilizes a supervised learning technique called back propagation for training [13]. For implementing the neural network model of a system, dataset of the input and output of the system is required. An application was implemented in C# for generating the desired input to be applied to the motor and recording the output of the DC motor simultaneously. Input data consists of sin wave, triangular wave, square wave with constant amplitude and square wave with random amplitude. This signal was generated by application that is implemented on a PC. The dataset are shown in Fig. 10.

4.5. Neural network (NN) structure

The multilayer perceptron neural network structure consist of five inputs and one output with one hidden layer are considered. The input of NN term consists of input of NN in time t and time t-1, outputs of NN at time t-1, t-2 and t-3. Structure of NN is obtained by genetic algorithm (GA) implementation. In this case, the number of NN hidden layers and the number of neurons in hidden layers are considered one to ten layers and neuron respectively. The GA cost function is the mean of squared of the errors (MSE) for different NN structures while the bias and weights are unit. GA results are shown in Table 8. It can be seen that the structure consists of one hidden layer with three neurons is the best choice.

Table 8. Parameters of motor.

Number of hidden layer	1	1	3
Number of neuron layer1	3	2	2
Number of neuron layer2	0	0	4
Number of neuron layer3	0	0	2
Number of neuron layer4	0	0	0
Number of neuron layer5	0	0	0
MSE	0.0047	0.0334	0.359

4.6. Neural Network training

In this paper the results are obtained for a controlled system (integration of the feed-forward controller and plant) in which the motor is controlled in a feed-forward loop using the proposed method and the plant is a propeller. However, the parameters introduced in Table 7, are actually the parameters of the motor and propeller together. NN is trained by LM method. Means of squared of the errors in this procedure is reached to 0.0001 in 15 epoch (Fig. 9). NN structure with five inputs, one hidden layer with three neurons and one output, is obtained and trained. The weights of connections are shown in equations (12) and (13).

$$w1 = \begin{bmatrix} -0.1216 & +0.3120 & +0.0666 \\ -0.7061 & +0.0650 & +0.0861 \\ -0.5052 & +0.0981 & +0.0456 \\ +0.0236 & -0.4529 & -0.1233 \\ +0.0852 & -2.0155 & +1.2733 \end{bmatrix} \quad (12)$$

$$w2 = \begin{bmatrix} -0.1151 \\ -0.4811 \\ +2.7612 \end{bmatrix} \quad (13)$$

The NN output with desired input is compared with experimental results and is shown in Fig. 11. The error between NN output and experimental output of the motor is shown in Fig. 12. It can be observed that the neural network output is the same

as the real output of the motor. The error between NN output and experiment output is less than 2 deg/s. So, this result is better than the parametric modelling of the DC motor. This result is contributed to the fact that in the NN model of a DC motor any linear and nonlinear effects of a DC motor are considered, while the parametric model of the motor only considers linear and static effects of a DC motor. So it can be concluded that the motor which is controlled using the proposed algorithm can successfully control the speed and torque of the propeller.

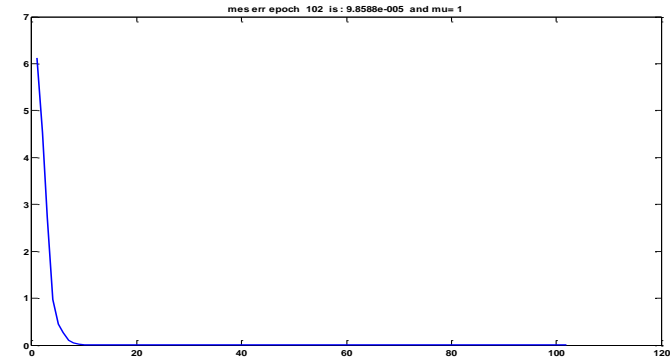


Figure 9. Mean squared normalized of NN for each epoch.

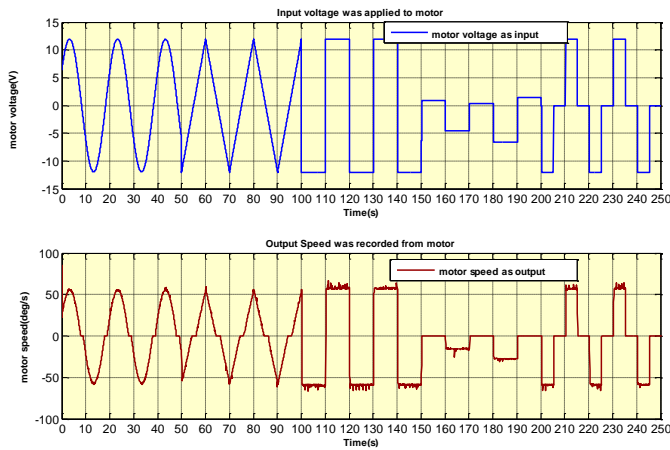


Figure 10. Experimental input (up) and output of the motor (down).

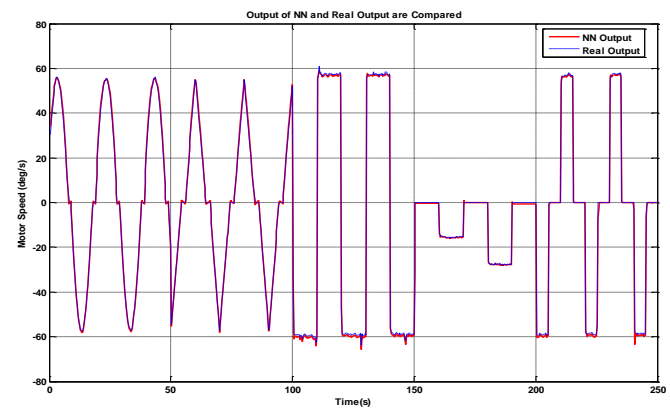


Figure 11. Experimental output compared to NN output.

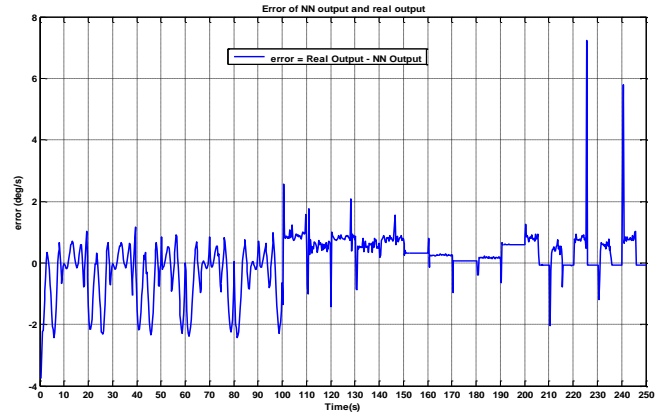


Figure 12. Error of experimental output and NN output.

Also comparing the proposed method with the simple traditional PID which is used for example in [14] shows that unlike PID algorithm which has a short delay and overshoot for tracking the desired speed and torque of the motor, the proposed algorithm of this paper is able to control the speed and torque of the propeller motor simultaneously and without the mentioned delay and overshoot as a result of its feedforward nature. Also in this method no sensor is required to be used in closed loop of PID. In [14], particle swarm optimization was used to cover this issue. Here the proposed structure in our paper proposes a way to control a DC motor without overshoot and with additional feed-forward loop to deal with torque or speed control.

5. Conclusions

In this paper, parametric model of a DC motor of the propeller of an aircraft was extracted by experimental test. This model is required for controlling the rotational velocity of the aircraft propeller using its inverse dynamics based on computed torque method. DC motor parameters can be affected by environmental condition, so this model can produce error in controlling procedure as a result of parametric uncertainties and nonlinear effect. For controlling the model with this uncertainty, closed loop PID controller was implemented. Neural network model of the motor was also extracted which covered all of the known and unknown parameters of the motor and was extremely useful for implementing computer based controllers. Levenberg-Marquardt algorithms which is one of the best methods of training a neural network was implemented in this research. Output of two proposed approached are compared and analyzed. In this comparison the motor is controlled using the proposed method in a feed-forward way and the plant is a propeller. It was shown that neural network model of the motor was more efficient. The neural network can be trained in an online way by implementing online training algorithms. Finally, it was concluded that NN can provide a more accurate model of a DC motor in the control procedure of feedback linearization methodology of an aircraft propellers. It was also discussed that although the proposed controlling algorithm has some advantages over traditional controlling methods like PID. Although PID is very effective and common in DC motor control either position or speed, though it has a drawback which is sensitivity to gain selection and improper selection of gains might lead to overshoot and this deficiency do not exist in the proposed algorithm.

References

- [1] B. W. McCormick, 1995, *Aerodynamics, aeronautics, and flight mechanics*, Wiley New York,
- [2] M. Selig, Modeling propeller aerodynamics and slipstream effects on small UAVs in realtime, in *Proceeding of*, 7938.
- [3] Ø. N. Smogeli, *Control of marine propellers: From normal to extreme conditions*, 2006.
- [4] C. Urrea, J. Kern, A new model for analog servo motors. Simulations and experimental results, *Canadian Journal on Automation, Control and Intelligent Systems*, Vol. 2, No. 2, pp. 29-38, 2011.
- [5] M. Anandaraju, P. Puttaswamy, J. Rajpurohit, Genetic Algorithm: An approach to Velocity Control of an Electric DC Motor, *International Journal of Computer Applications*, Vol. 26, No. 1, pp. 37-43, 2011.
- [6] K. Ohishi, M. Nakao, K. Ohnishi, K. Miyachi, Microprocessor-controlled DC motor for load-insensitive position servo system, *IEEE Transactions on Industrial Electronics*, No. 1, pp. 44-49, 1987.
- [7] M. Hoque, M. Zaman, M. Rahman, Artificial neural network based controller for permanent magnet dc motor drives, in *Proceeding of, IEEE*, pp. 1775-1780.
- [8] J. Liu, P. Zhang, F. Wang, Real-time dc servo motor position control by pid controller using labview, in *Proceeding of, IEEE*, pp. 206-209.
- [9] Q.-x. Zhu, Real-time DC motor position control by (FPID) controllers and design (FLC) using labview software simulation, in *Proceeding of, IEEE*, pp. 417-420.
- [10] D. Yan-hong, W. Yong, S. Hui-Yong, L. Hua, PID controller optimization of mobile robot servo system, in *Proceeding of, IEEE*, pp. 235-237.
- [11] R. Krishnan, 2001, *Electric motor drives: modeling, analysis, and control*, Prentice hall New Jersey,
- [12] M. T. Hagan, M. B. Menhaj, Training feedforward networks with the Marquardt algorithm, *IEEE transactions on Neural Networks*, Vol. 5, No. 6, pp. 989-993, 1994.
- [13] F. Rosenblatt, *Principles of neurodynamics. perceptrons and the theory of brain mechanisms*, CORNELL AERONAUTICAL LAB INC BUFFALO NY, pp. 1961.
- [14] B. Allaoua, B. Gasbaoui, B. Mebarki, Setting up PID DC motor speed control alteration parameters using particle swarm optimization strategy, *Leonardo Electronic Journal of Practices and Technologies*, Vol. 14, pp. 19-32, 2009.