



Original Article

Back-stepping Control of Switched Reluctance Motor with Artificial Neural Network based Flux Estimator

Phi Hoang Nha^{1,2}, Pham Hung Phi², Dao Quang Thuy³, Le Xuan Hai¹,
Pham Xuan Dat², Nguyen Ngoc Linh^{4,*}

¹Ha Noi University of Industry, 298 Cau Dien, Bac Tu Liem, Ha Noi, Vietnam

²Hanoi University of Science and Technology, 1 Dai Co Viet, Hai Ba Trung, Ha Noi, Vietnam

³Ministry of Science and Technology, 133 Tran Duy Hung, Cau Giay, Hanoi, Vietnam

⁴VNU University of Engineering and Technology, 144 Xuan Thuy, Cau Giay, Hanoi, Vietnam

Received 28 July 2021

Revised 01 October 2021; Accepted 09 October 2021

Abstract: The paper presents a new approach to the speed control of a switched reluctance motors (SRM) is that using a back-stepping controller combining with an artificial neuron network based flux estimator. The nonlinear mathematical model of switched reluctance motor (SRM) is established and the back-stepping control strategy is applied to control SRM. The ANN will be used to estimate the flux of the motor instead of approximated model or experimental values. The ANN flux estimator was trained off-line using backpropagation algorithm. The stability of the closed-loop control system was analyzed and proved according to the Lyapunov stability criteria. The simulation is carried out with both traditional back-stepping controller and the back-stepping controller combining with ANN based flux estimator. The numerical simulation results confirmed quality of the back-stepping controller as well as the feasibility of using ANN in the flux estimator.

Keywords: Switched Reluctance Motor (SRM), Back-stepping Control, Flux Estimator, Artificial Neural Networks (ANN), Backpropagation Algorithm.

1. Introduction

The switched reluctance motors (SRMs) is more and more widely used in variable speed drives thanks to many advantages such as control flexibility, simple structure, lower cost and high

efficiency, etc. The rotor has no wire allow it withstand high temperature as well as is suitable for extremely high speed application. However, SRM has non-linear characteristic due to magnetic saturation, which makes it difficult to control its torque [1, 5, 7]. Control of SRM

* Corresponding author.

E-mail address: nnguyen@vnu.edu.vn

<https://doi.org/10.25073/2588-1086/vnucsce.261>

therefore is a challenging problem and depends much on the mathematic model of SRM. Many studies have tried to establish the nonlinear mathematical model of SRMs [1-6, 8, 13-14]. Some mathematical models have been developed, however, their coefficients are difficult to determine (because depending on the type of SRM, the size of SRM...[1, 8]. In [5], author introduced a non-linear model of SRM and did a linearization after that. Besides, one of the most important components in SRM's model, used in controller design, is the motor's magnetic flux. Some methods have been proposed to estimate motor's flux of SRMs based on the experimental results or using an approximately mathematical model [4, 14] However, these methods usually have difficulties in reality and sometimes can make a large error. To overcome this drawback, in this research, we will use an artificial neuron network (ANN) to estimate the flux of SRMs.

Accordingly, non-linear model of the switched reluctance motor, which including the phase switch and the dynamics of the SRMs, has been established and been used to synthesize the back-stepping controller. In this model, motor's flux will be estimated by ANN based flux estimator instead of using approximate model or measured data. This ANN is trained offline based on the experimental data and can continue automatically be trained in the future.. In order to verify the efficiency and feasibility of proposed method, several simulations are implemented both for back-stepping control (BTP) with traditional model of SRM and back-stepping control with the model using the ANN based flux estimator (BTP - ANN flux estimator). Conclusions and some future works are pointed out in final.

2. Mathematic Model of the SRMS

Mathematical model of the m-phase SRMs is constructed from the basic machine equations

including armature voltage equation, electro-magnetic torque equation and mechanical equation (1):

$$\begin{cases} u_j = R.i_j + \frac{d\psi_j}{dt} \\ T_j(\theta, i_j) = \frac{\partial W'_j}{\partial \theta} \\ J \frac{d^2\theta}{dt^2} = T_e - T_l \end{cases} \quad (1)$$

in which $j = 1, 2, \dots, m$

u_j is voltage of phase j

R is resistor of phase j

i_j is current of phase j

ψ_j is flux of phase j

T_e is torque of phase

T_l is torque of load

J is moment of inertia

W'_j is the electro-magnetic energy

which is determined in (2):

$$\partial W'_j(\theta, i_j) = \int_0^{i_j} \psi_j(\theta, i_j) di_j \quad (2)$$

Electrical torque in SRM is a nonlinear function of only current if the magnetic circuit is linear.

The total torque produced is equal to the sum of the moments in the phases:

$$T_e(\theta, i_1, i_2, \dots, i_m) = \sum_{j=1}^m T_j(\theta, i_j) \quad (3)$$

To control the SRMs, we need to determine the magnetic flux characteristic $\psi_j(\theta, i_j)$ as accurately as possible. For convenience in the process of research and development of control algorithms, the function of magnetic flux characteristic in [4] will be used:

$$\psi_j(\theta, i_j) = \psi_s(1 - e^{-i_j f_j(\theta)}) \quad (4)$$

with $j = 1, 2, \dots, m$; ψ_s is saturation flux.

If we ignore the higher order components in the Fourier series, we have a function $f_j(\theta)$:

$$f_j(\theta) = a + b \sin[N_r \theta - (j-1) \frac{2\pi}{m}] \quad (5)$$

N_r is number of rotor's pole.

Torque of phase j is represented as follows:

$$T_j(\theta, i_j) = \frac{\psi_s}{f_j^2(\theta)} \frac{df_j(\theta)}{d\theta} \{1 - [1 + i_j f_j(\theta)] e^{-i_j f_j(\theta)}\} \quad (6)$$

The state-space model of the SRMs can be obtained from the following equations:

$$\begin{cases} \frac{d\theta}{dt} = \omega \\ \frac{d\omega}{dt} = \frac{1}{J} \left\{ \sum_{j=1}^m T_j(\theta, i_j) - T_l(\theta, \omega) \right\} \\ \frac{di_j}{dt} = - \left(\frac{\partial \psi_j}{\partial i_j} \right) \left(Ri_j + \frac{\partial \psi_j}{\partial \theta} \omega \right) + \left(\frac{\partial \psi_j}{\partial i_j} \right)^{-1} u_j \end{cases} \quad (7)$$

Considering the switched reluctance motor with $m = 4$ phases, the state vector is $x = [\theta, \omega, i_1, i_2, i_3, i_4]^T = [x_1, x_2, x_3, x_4, x_5, x_6]^T$. The state-space equations of motor [13]:

$$\begin{aligned} \dot{x}_1 &= x_2 & (8) \\ \dot{x}_2 &= \frac{1}{J} \left[T_1(\theta, x_3) + T_2(\theta, x_4) + T_3(\theta, x_5) \right. \\ &\quad \left. + T_4(\theta, x_6) - T_l(x_1, x_2) \right] \\ &= \frac{1}{J} \left[\frac{\psi_s}{f_1^2(x_1)} \frac{\partial f_1(x_1)}{\partial x_1} N_r \{1 - [1 + x_3 f_1(x_1)] e^{-x_3 f_1(x_1)}\} \right. \\ &\quad \left. + \frac{\psi_s}{f_2^2(x_1)} \frac{\partial f_2(x_1)}{\partial x_1} N_r \{1 - [1 + x_4 f_2(x_1)] e^{-x_4 f_2(x_1)}\} \right. \\ &\quad \left. + \frac{\psi_s}{f_3^2(x_1)} \frac{\partial f_3(x_1)}{\partial x_1} N_r \{1 - [1 + x_5 f_3(x_1)] e^{-x_5 f_3(x_1)}\} \right. \\ &\quad \left. + \frac{\psi_s}{f_4^2(x_1)} \frac{\partial f_4(x_1)}{\partial x_1} N_r \{1 - [1 + x_6 f_4(x_1)] e^{-x_6 f_4(x_1)}\} \right. \\ &\quad \left. - Bx_2 - mgl \sin(x_1) \right] \\ \dot{x}_3 &= \left[-\psi_s e^{-x_3 f_1(x_1)} f_1(x_1) \right]^{-1} \left[Rx_3 + \left(\psi_s e^{-x_3 f_1(x_1)} \right) \right. \\ &\quad \left. \left(x_3 \frac{\partial f_1(x_1)}{\partial x_1} \right) x_2 \right] \\ &\quad + \left[\psi_s e^{-x_3 f_1(x_1)} f_1(x_1) \right]^{-1} u_1 \end{aligned} \quad (9)$$

$$\dot{x}_4 = \left[-\psi_s e^{-x_4 f_2(x_1)} f_2(x_1) \right]^{-1} \left[Rx_4 + \left(\psi_s e^{-x_4 f_2(x_1)} \right) \right. \\ \left. \left(x_4 \frac{\partial f_2(x_1)}{\partial x_1} \right) x_2 \right] \\ + \left[\psi_s e^{-x_4 f_2(x_1)} f_2(x_1) \right]^{-1} u_2 \quad (11)$$

$$\dot{x}_5 = \left[-\psi_s e^{-x_5 f_3(x_1)} f_3(x_1) \right]^{-1} \left[Rx_5 + \left(\psi_s e^{-x_5 f_3(x_1)} \right) \right. \\ \left. \left(x_5 \frac{\partial f_3(x_1)}{\partial x_1} \right) x_2 \right] \\ + \left[\psi_s e^{-x_5 f_3(x_1)} f_3(x_1) \right]^{-1} u_3 \quad (12)$$

$$\dot{x}_6 = \left[-\psi_s e^{-x_6 f_4(x_1)} f_4(x_1) \right]^{-1} \left[Rx_6 + \left(\psi_s e^{-x_6 f_4(x_1)} \right) \right. \\ \left. \left(x_6 \frac{\partial f_4(x_1)}{\partial x_1} \right) x_2 \right] \\ + \left[\psi_s e^{-x_6 f_4(x_1)} f_4(x_1) \right]^{-1} u_4 \quad (13)$$

where:

$$\frac{\partial f_i}{\partial x_1} = b N_r \cos \left(N_r x_1 - (j-1) \frac{2\pi}{m} \right) \quad (14)$$

In (9), we set:

$$\begin{aligned} f_a(x) &= \frac{1}{J} \left[\frac{\psi_s}{f_1^2(x_1)} \frac{\partial f_1(x_1)}{\partial x_1} N_r \{1 - e^{-x_3 f_1(x_1)}\} \right] \\ g_a(x) &= \frac{1}{J} \left[\frac{\psi_s}{f_1^2(x_1)} \frac{\partial f_1(x_1)}{\partial x_1} N_r \{-f_1(x_1) e^{-x_3 f_1(x_1)}\} \right] \\ f_b(x) &= \frac{1}{J} \left[\frac{\psi_s}{f_2^2(x_1)} \frac{\partial f_2(x_1)}{\partial x_1} N_r \{1 - e^{-x_4 f_2(x_1)}\} \right] \\ g_b(x) &= \frac{1}{J} \left[\frac{\psi_s}{f_2^2(x_1)} \frac{\partial f_2(x_1)}{\partial x_1} N_r \{-f_2(x_1) e^{-x_4 f_2(x_1)}\} \right] \\ f_c(x) &= \frac{1}{J} \left[\frac{\psi_s}{f_3^2(x_1)} \frac{\partial f_3(x_1)}{\partial x_1} N_r \{1 - e^{-x_5 f_3(x_1)}\} \right] \\ g_c(x) &= \frac{1}{J} \left[\frac{\psi_s}{f_3^2(x_1)} \frac{\partial f_3(x_1)}{\partial x_1} N_r \{-f_3(x_1) e^{-x_5 f_3(x_1)}\} \right] \\ f_d(x) &= \frac{1}{J} \left[\frac{\psi_s}{f_4^2(x_1)} \frac{\partial f_4(x_1)}{\partial x_1} N_r \{1 - e^{-x_6 f_4(x_1)}\} \right] \\ g_d(x) &= \frac{1}{J} \left[\frac{\psi_s}{f_4^2(x_1)} \frac{\partial f_4(x_1)}{\partial x_1} N_r \{-f_4(x_1) e^{-x_6 f_4(x_1)}\} \right] \end{aligned}$$

Equation (9) can be rewritten as follows:

$$\begin{aligned} \dot{x}_2 = & [f_a(x) + g_a(x)x_3] + [f_b(x) + g_b(x)x_4] \\ & + [f_c(x) + g_c(x)x_5] + [f_d(x) + g_d(x)x_6] \\ & - \frac{B}{J}x_2 - \frac{mgl}{J}\sin(x_1) \end{aligned} \quad (15)$$

Differentiating equation (16), we have:

$$\begin{aligned} \ddot{x}_2 = & [\dot{f}_a(x) + \dot{g}_a(x)x_3 + g_a(x)\dot{x}_3] \\ & + [\dot{f}_b(x) + \dot{g}_b(x)x_4 + g_b(x)\dot{x}_4] \\ & + [\dot{f}_c(x) + \dot{g}_c(x)x_5 + g_c(x)\dot{x}_5] \\ & + [\dot{f}_d(x) + \dot{g}_d(x)x_6 + g_d(x)\dot{x}_6] \\ & - \frac{B}{J}\dot{x}_2 - \frac{mgl}{J}\cos(x_1)\dot{x}_1 \end{aligned} \quad (16)$$

From (11) to (14), we set:

$$p_a(x) = \left[-\psi_s e^{-x_3 f_1(x_1)} f_1(x_1) \right]^{-1} \begin{bmatrix} Rx_3 \\ + (\psi_s e^{-x_3 f_1(x_1)}) \\ \left(x_3 \frac{\partial f_1(x_1)}{\partial x_1} \right) x_2 \end{bmatrix}$$

$$q_a(x) = \left[\psi_s e^{-x_3 f_1(x_1)} f_1(x_1) \right]^{-1}$$

$$p_b(x) = \left[-\psi_s e^{-x_4 f_2(x_1)} f_2(x_1) \right]^{-1} \begin{bmatrix} Rx_4 \\ + (\psi_s e^{-x_4 f_2(x_1)}) \\ \left(x_4 \frac{\partial f_2(x_1)}{\partial x_1} \right) x_2 \end{bmatrix}$$

$$q_b(x) = \left[\psi_s e^{-x_4 f_2(x_1)} f_2(x_1) \right]^{-1}$$

$$p_c(x) = \left[-\psi_s e^{-x_5 f_3(x_1)} f_3(x_1) \right]^{-1} \begin{bmatrix} Rx_5 \\ + (\psi_s e^{-x_5 f_3(x_1)}) \\ \left(x_5 \frac{\partial f_3(x_1)}{\partial x_1} \right) x_2 \end{bmatrix}$$

$$q_c(x) = \left[\psi_s e^{-x_5 f_3(x_1)} f_3(x_1) \right]^{-1}$$

$$p_d(x) = \left[-\psi_s e^{-x_6 f_4(x_1)} f_4(x_1) \right]^{-1} \begin{bmatrix} Rx_6 + \\ \left(\psi_s e^{-x_6 f_4(x_1)} \right) \\ \left(x_6 \frac{\partial f_4(x_1)}{\partial x_1} \right) x_2 \end{bmatrix}$$

$$q_d(x) = \left[\psi_s e^{-x_6 f_4(x_1)} f_4(x_1) \right]^{-1}$$

Rewritten equations from (10) to (13) as follows:

$$\begin{aligned} \dot{x}_3 &= p_a(x) + q_a(x)u_1 \\ \dot{x}_4 &= p_b(x) + q_b(x)u_2 \\ \dot{x}_5 &= p_c(x) + q_c(x)u_3 \\ \dot{x}_6 &= p_d(x) + q_d(x)u_4 \end{aligned} \quad (17)$$

Substituting (17) to (16), we have:

$$\begin{aligned} \ddot{x}_2 = & \left[\dot{f}_a(x) + \dot{g}_a(x)x_3 + g_a(x)p_a(x) \right] \\ & + \left[g_a(x)q_a(x)u_1 \right] \\ & + \left[\dot{f}_b(x) + \dot{g}_b(x)x_4 + g_b(x)p_b(x) \right] \\ & + \left[g_b(x)q_b(x)u_2 \right] \\ & + \left[\dot{f}_c(x) + \dot{g}_c(x)x_5 + g_c(x)p_c(x) \right] \\ & + \left[g_c(x)q_c(x)u_3 \right] \\ & + \left[\dot{f}_d(x) + \dot{g}_d(x)x_6 + g_d(x)p_d(x) \right] \\ & + \left[g_d(x)q_d(x)u_4 \right] \\ & - \frac{B}{J}\dot{x}_2 - \frac{mgl}{J}\cos(x_1)\dot{x}_1 \end{aligned} \quad (18)$$

The switched reluctance motor works with the principle of voltage supply for each phase. If the number of phases is 4, we have $u_j = k_j \mu$, with $j=1, 2, 3, 4$; k_j is a phase transition key, so it can only take 2 values, 0 or 1. Equation (18) can be represented as follows:

$$\ddot{x}_2 = \begin{bmatrix} \dot{f}_a(x) + \dot{g}_a(x)x_3 + g_a(x)p_a(x) + \dot{f}_b(x) \\ + \dot{g}_b(x)x_4 + g_b(x)p_b(x) + \dot{f}_c(x) + \\ \dot{g}_c(x)x_5 + g_c(x)p_c(x) + \dot{f}_d(x) \\ + \dot{g}_d(x)x_6 + g_d(x)p_d(x) \end{bmatrix} + \begin{bmatrix} g_a(x)q_a(x)k_1 + g_b(x)q_b(x)k_2 \\ + g_c(x)q_c(x)k_3 + g_d(x)q_d(x)k_4 \end{bmatrix} - \frac{B}{J}\dot{x}_2 - \frac{mgl}{J}\cos(x_1)\dot{x}_1 \quad (19)$$

We consider:

$$F(x) = \begin{bmatrix} \dot{f}_a(x) + \dot{g}_a(x)x_3 + g_a(x)p_a(x) + \dot{f}_b(x) \\ + \dot{g}_b(x)x_4 + g_b(x)p_b(x) + \dot{f}_c(x) + \\ \dot{g}_c(x)x_5 + g_c(x)p_c(x) + \dot{f}_d(x) \\ + \dot{g}_d(x)x_6 + g_d(x)p_d(x) \end{bmatrix}$$

and:

$$G(x) = \begin{bmatrix} g_a(x)q_a(x)k_1 + g_b(x)q_b(x)k_2 \\ + g_c(x)q_c(x)k_3 + g_d(x)q_d(x)k_4 \end{bmatrix}$$

We have another expression of equation (19) as follows:

$$\ddot{x}_2 = F(x) + G(x) - \frac{B}{J}\dot{x}_2 - \frac{mgl}{J}\cos(x_1)\dot{x}_1 \quad (20)$$

If we set:

$$f(x) = F(x) - \frac{B}{J}\dot{x}_2 - \frac{mgl}{J}\cos(x_1)\dot{x}_1 \quad (21)$$

$$g(x) = G(x)$$

We have:

$$\ddot{x}_2 = f(x) + g(x)u \quad (22)$$

Setting $x_2 = z_1$, we have the state model of the SRMs as presented in (23):

$$\begin{cases} \dot{z}_1 = z_2 \\ \dot{z}_2 = f(x) + g(x)u \end{cases} \quad (23)$$

with $f(x), g(x)$ is defined in equation (21).

This is a second order tight backpropagation model. Model in (23) is perfectly suitable using back-stepping technique to design the controller.

3. Design a Back-Stepping Controller Combining with ANN based Flux Estimator for SRM

3.1. The Back-stepping Controller

As mention above, the model of SRM in (23) is a second order tight backpropagation type. According to the back-stepping technique, we perform two design steps for this system [14][15].

Step 1: Considering the error tracking reference velocity $z_d = \omega_d$ is e_1 , we have:

$$e_1 = z_1 - z_d \quad (24)$$

Differentiating e_1 over time, we have:

$$\dot{e}_1 = \dot{z}_1 - \dot{z}_d = z_2 - \dot{z}_d \quad (25)$$

Considering $e_2 = z_2 - \alpha_1$ where α_1 is virtual control signal for the first subsystem.

Substitution to (25), we have:

$$\dot{e}_1 = \dot{z}_1 - \dot{z}_d = z_2 - \dot{z}_d = e_2 + \alpha_1 - \dot{z}_d \quad (26)$$

To determine the virtual control signal to ensure $e_1 \rightarrow 0$ we choose Lyapunov function:

$$V_1 = \frac{1}{2}e_1^2 \quad (27)$$

Differentiating V_1 over time, we have:

$$\dot{V}_1 = e_1\dot{e}_1 = e_1(e_2 + \alpha_1 - \dot{z}_d) = -c_1e_1^2 + e_1e_2 \quad (28)$$

In order to have (30), the virtual control signal has a following form:

$$\alpha_1 = -c_1e_1 + \dot{z}_d \quad (29)$$

in which c_1 is positive constant. To $e_1 \rightarrow 0$ then $e_2 \rightarrow 0$.

$$\text{Step 2: } e_2 = z_2 - \alpha_1 \quad (30)$$

Differentiating e_2 over time, we have:

$$\dot{e}_2 = \dot{z}_2 - \dot{\alpha}_1 \tag{31}$$

From (25) we have:

$$\dot{e}_2 = \dot{z}_2 - \dot{\alpha}_1 = f(x) + g(x)u - \dot{\alpha}_1 \tag{32}$$

To determine control signal u to ensure $e_2 \rightarrow 0$, we also choose Lyapunov function:

$$V_2 = V_1 + \frac{1}{2}e_2^2 \tag{33}$$

Differential over time, we have:

$$\dot{V}_2 = \dot{V}_1 + e_2\dot{e}_2 \tag{34}$$

Substituting equations (28) and (33) to (34), we have:

$$\dot{V}_2 = -c_1e_1^2 + e_1e_2 + e_2[f(x) + g(x)u - \dot{\alpha}_1] \tag{35}$$

Selecting the control signal of system from (35):

$$u = \frac{-c_2e_2 - e_1 - [f(x) - \dot{\alpha}_1]}{g(x)} \tag{36}$$

with c_2 is positive constant.

Theorem: The SRM has been represented in state-space model (23) controlled by a back-stepping controller defined in (36) where are positive constants to ensure a stable Lyapunov closed system.

Proof: We choose the Lyapunov function for closed loop system has a following form:

$$V = \frac{1}{2}(e_1^2 + e_2^2) = V_1 + \frac{1}{2}e_2^2 = V_2 \tag{37}$$

Differential V over time, we have:

$$\dot{V} = -c_1e_1^2 + e_1e_2 + e_2[f(x) + g(x)u - \dot{\alpha}_1] \tag{38}$$

Substituting u from equation (36) to (38), we have:

$$\dot{V} = -c_1e_1^2 + e_1e_2 + e_2 \begin{bmatrix} f(x) - c_2e_2 - e_1 \\ -[f(x) - \dot{\alpha}_1] - \dot{\alpha}_1 \end{bmatrix} \tag{39}$$

$$\dot{V} = -c_1e_1^2 - c_2e_2^2 \leq 0$$

→QED.

3.2. ANN Based Flux Estimator for SRM

Based on magnetic flux equation (4) of SRM, we will approximate function $f_j(\theta)$ by an ANN. The structure of ANN is illustrated in Figure 1 [10]. Inputs of the network are the current in stator and position of rotor, and outputs are $f_j(\theta)$ and $\hat{\psi}_j(\theta)$. The network includes 5 layers P^f, R^f, S^f, T^f, U^f with the letters representing the relative number of neurons in respective layers f (forward) is in every variable of the network.

i) Feedforward algorithm of the estimator

In every layer, x is the input and y is the output of each neuron. Signal propagation and activation function is illustrated.

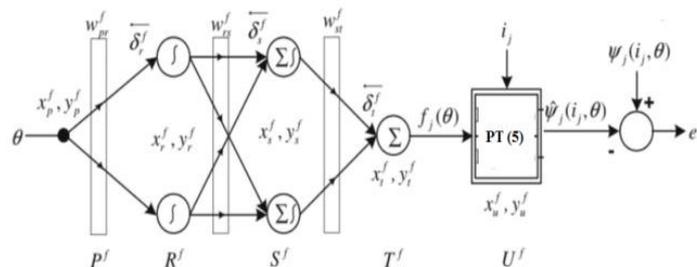


Figure 1. Neuron network structure of the system.

Layer P^f : neuron p has input and output presented:

$$x_p^f = \theta \text{ and } y_p^f(x_p^f) = x_p^f \quad (40)$$

with $p=0$. Activation function equals to 1 in this layer.

Layer R^f : Each neuron r in this layer has input and output presented:

$$x_r^f = y_p^f \cdot w_{pr}^f \text{ and } y_r^f = \exp\left(-\left(\frac{x_r^f - c_r^f}{\sigma_r^f}\right)^2\right) \quad (41)$$

with $p=0$ and $r=0\dots R^f$. In which c_r^f and σ_r^f are center and range of Gaussian Activation function. w_{pr}^f is the weight between 2 layers P^f and R^f .

Layer S^f : s neurons are in this layer. Inputs and outputs are calculated:

$$x_s^f = \sum_{r=0}^{R^f} y_r^f \cdot w_{rs}^f \quad (42)$$

and

$$y_s^f = \exp\left(-\left(\frac{x_s^f - c_s^f}{\sigma_s^f}\right)^2\right) \quad (43)$$

with $r=0\dots R^f$ and $s=0\dots S^f$. In which c_s^f and σ_s^f are center and range of Gaussian Activation function. w_{rs}^f is the weight between 2 layers R^f and S^f .

Layer T^f : Function $f_j(\theta)$ is the output of this layer. For each neuron t in this layer is determined as:

$$x_t^f = \sum_{r=0}^{R^f} y_r^f \cdot w_{rt}^f \text{ and } y_t^f = x_t^f = f_j(\theta) \quad (44)$$

with $s=0\dots S^f$ and $t=0$.

Layer U^f : Magnetic flux and torque of each phase are determined in this layer. After obtaining $f_j(\theta)$ and $a_j(\theta)$ from the output of T^f , approximated magnetic flux function $\hat{\psi}_j(\theta)$ and approximated torque function $\hat{T}_j(\theta, i_j)$ are calculated in layer U^f according to the formula:

$$x_u^f = y_t^f i_j \text{ and } y_u^f = 1 - e^{-x_u^f} = \hat{\psi}_j(\theta) \quad (45)$$

with $u=0$. In which i_j is the current in stator windings.

Electro-magnetic torque is approximated as:

$$\hat{T}_j(\theta, i_j) = \frac{df_j(\theta)}{f_j^2(\theta)d\theta} [1 - (1 + i_j f_j(\theta))e^{-i_j f_j(\theta)}] \quad (46)$$

In above formulas, we do not need to consider the saturation of flux ψ_s because ANN has adaptive structure through weights and activation functions. To train the ANN, backpropagation algorithm is used.

ii). Backpropagation algorithm

In ANN, the main purpose of network training is to update the network's weights (Figure 2). The algorithm for training the ANN with the forward model is error backpropagation algorithm. Flowchart of training algorithm is presented in Figure 3.

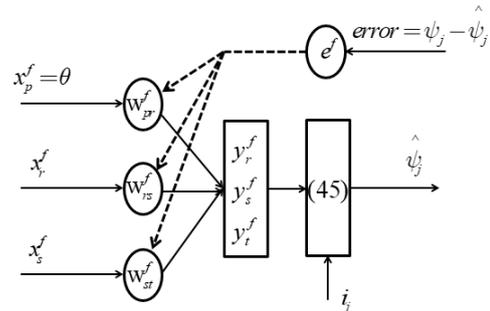


Figure 2. Neural network training process

Weights of the network are trained to minimize an objective function:

$$E^f = \frac{1}{2} (e^f(k))^2 \text{ with } k=1, \dots, K^f \quad (47)$$

in which K^f is the number of the input and output and e^f is the error between approximated value of magnetic flux $\hat{\psi}_j$ and actual value ψ_j .

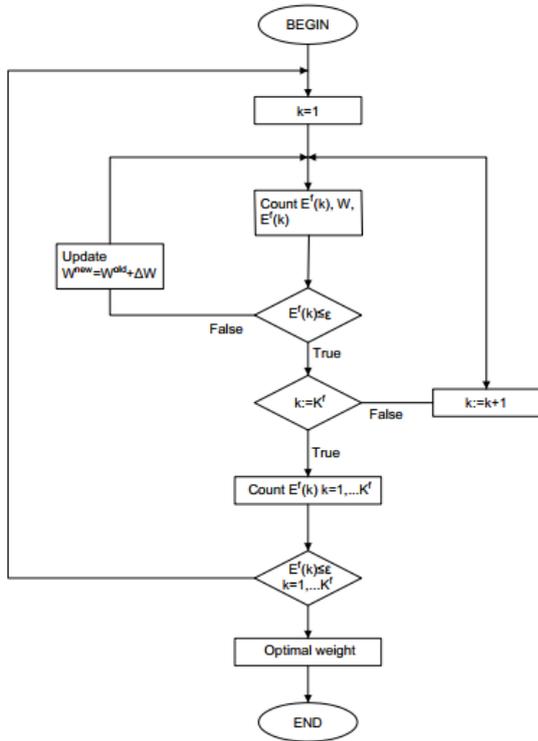


Figure 3. Diagram of neural network training algorithm

$$e^f = \psi_j(k) - \hat{\psi}_j(k) \text{ with } k = 1, \dots, K^f \quad (48)$$

Layer $U^f - T^f$: Because weights between layers are uniform, e^f is directly fed into T^f in chain rule. Therefore, error $\bar{\delta}_t^f$ determined:

$$\bar{\delta}_t^f = -e^f i_j \frac{\partial y_u^f}{\partial x_u^f} \frac{\partial y_t^f}{\partial x_t^f} \quad (49)$$

Layer $T^f - S^f$: In this layer, the weights change:

$$\Delta w_{st}^f = \eta_{st}^f \left(-\frac{\partial E^f}{\partial w_{st}^f} \right) = \eta_{st}^f \bar{\delta}_t^f y_s^f \quad (50)$$

in which η_{st}^f is the learning coefficient of the weight between.

Layer $S^f - R^f$: Error e^f is directly fed S^f by chain rule. Therefore, error $\bar{\delta}_s^f$ is determined:

$$\bar{\delta}_s^f = \bar{\delta}_s^f w_{st}^f \frac{\partial y_s^f}{\partial x_s^f} \quad (51)$$

In this layer, the change of the weight is:

$$\Delta w_{rs}^f = \eta_{rs}^f \left(-\frac{\partial E^f}{\partial w_{rs}^f} \right) = \eta_{rs}^f \bar{\delta}_s^f y_r^f \quad (52)$$

in which η_{rs}^f is the learning coefficient of the weight between 2 layers.

Layer $R^f - P^f$: Error e^f is directly fed S^f by chain rule. Therefore, error $\bar{\delta}_r^f$ is determined:

$$\bar{\delta}_r^f = \frac{\partial y_r^f}{\partial x_r^f} \sum_{s=0}^{S^f} \bar{\delta}_s^f w_{rs}^f \quad (53)$$

In this layer, the change of the weight is:

$$\Delta w_{pr}^f = \eta_{pr}^f \left(-\frac{\partial E^f}{\partial w_{pr}^f} \right) = \eta_{pr}^f \bar{\delta}_r^f y_p^f \quad (54)$$

in which η_{pr}^f is the learning coefficient of the weight between 2 layers.

Weights w_{pr}^f, w_{rs}^f and w_{st}^f are updated through energy function E^f . The change in weights $\Delta w_{st}^f, \Delta w_{rs}^f, \Delta w_{pr}^f$ will be added to weights in the ANN as in (55):

$$\begin{aligned}
 w_{st}^f(k+1) &= w_{st}^f(k) + \Delta w_{st}^f \\
 w_{rs}^f(k+1) &= w_{rs}^f(k) + \Delta w_{rs}^f \\
 w_{pr}^f(k+1) &= w_{pr}^f(k) + \Delta w_{pr}^f
 \end{aligned} \quad (55)$$

with $k = 1, \dots, K^f$.

The back-stepping controller proposed is only possible when the state variables of the SRM are provided. The flux state variable with parameters that are difficult to determine is provided from the estimator in section 3.2. Back-stepping control technique (36) for SRM that combines magnetic flux estimator by neural network is proposed. The neural network, after being trained offline, is fed to the controller as shown in Figure 4.

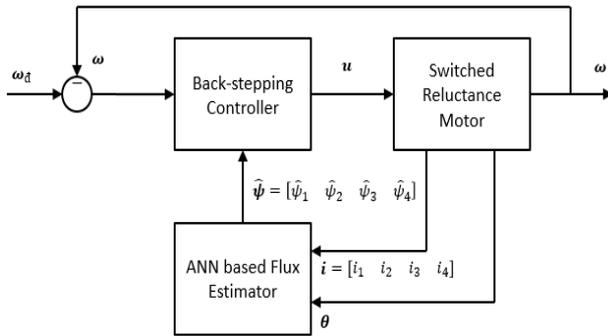


Figure 4. The back-stepping controller combined with ANN based flux estimator.

4. Simulation Result

The proposed control system in the paper is verified by the simulation results carried out through Matlab/SIMULINK software.

The design criteria for this problem are:

- o No static error.
- o Overshoot less than 5%.
- o Settling time less than 0.5s.

The parameters of the neural network after being trained, the SRM parameters and the selected parameters of the controller in Table 1.

Training neural network parameters of flux estimator:

$$\begin{aligned}
 R^f &= S^f = 20, \quad K^f = 200, \\
 \sigma_r^f &= \text{linspace}(-5, 5, R^f), \\
 \sigma_s^f &= \text{linspace}(-5, 5, S^f), \quad c_r^f = c_s^f = 0.1, \\
 \eta_{st}^f &= \eta_{rs}^f = \eta_{pr}^f = 0.02
 \end{aligned}$$

Table 1. Parameters of SRM and controller:

| | |
|---|----------------|
| $N_r = 6$ | $c_1 = 2$ |
| $J = 6.8 \times 10^3 \text{ (kg / m}^2\text{)}$ | $c_2 = 0.1$ |
| $R = 0.05 \text{ (}\Omega\text{)}$ | $\gamma = 100$ |
| $a = 1.5 \times 10^{-3} \text{ (H)}$ | $T = 0.025$ |
| $b = 1.364 \times 10^{-3} \text{ (H)}$ | $l_1 = 100$ |
| $B = 0.2$ | $l_2 = 2500$ |
| $l = 2 \text{ (m)}$ | |

Simulation results of the performance of the proposed control system are shown in Figure 5, Figure 6, Figure 7 and Figure 8

In Figure 5, the approximated magnetic flux from the ANN based flux estimator is compared with its values calculated by approximated mathematic model (Figure 5a). The error of the two values is shown in Figure 5b. It can be seen that, the value of the error is nearly zero. It is proved that the ANN work well.

In Figure 6, the electromagnetic torque of SRM is presented. It is clearly that the ripple still exists. This problem usually appears with SRM and need to be improve in this research.

We continue considering the performance of control system. According to this, the back-stepping controller (BTP) and the back-stepping controller using ANN based flux estimator (BTP-ANN flux estimator) are used to control the speed for the SRM (Figures 7, 8) with the same conditions.

In Figure 7, we consider the response of system at a fixed set point at 10 rad/s. The performance of the two simulations is compared and summarized in Table 2. In Figure 8, we continue verifying the performance of system when the system has been change in operation. In detail, at time $t = 1$ s, the set point change from 15 rad/s to 20 rad/s. We can see that, system still tracks the set point.

Table 2. Control performance between BTP and BTP-ANN flux estimator:

| | BTP | BTP-ANN flux estimator |
|----------------------|-----------|------------------------|
| Static error (rad/s) | 10^{-4} | 10^{-4} |
| Setting time (s) | 0.45 | 0.45 |
| Overshoot (%) | 0 | 0 |

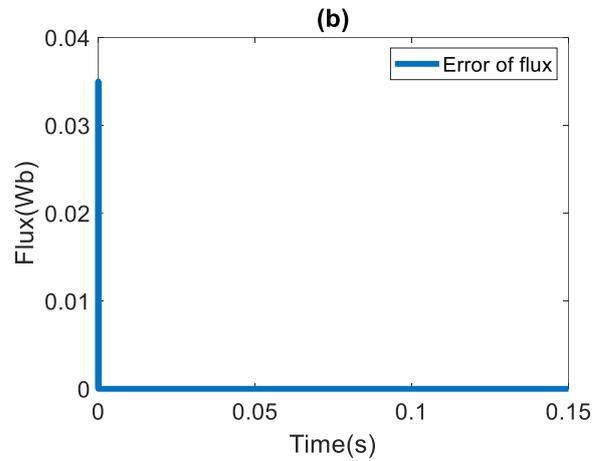
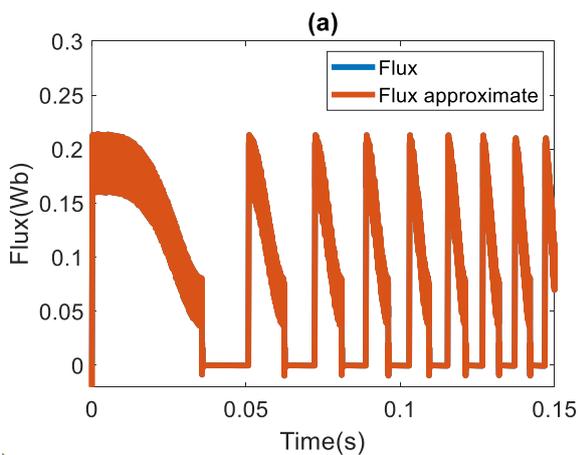


Figure 5. Magnetic flux characteristic.

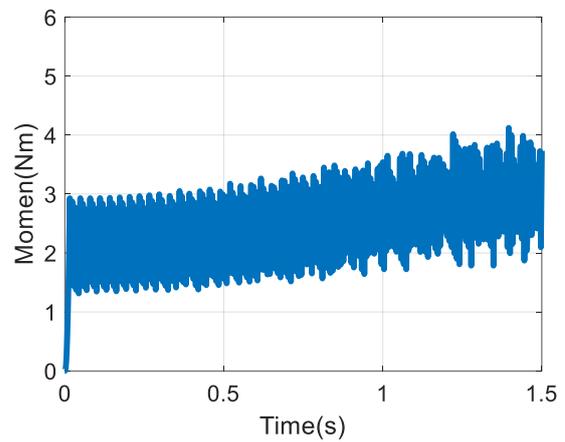
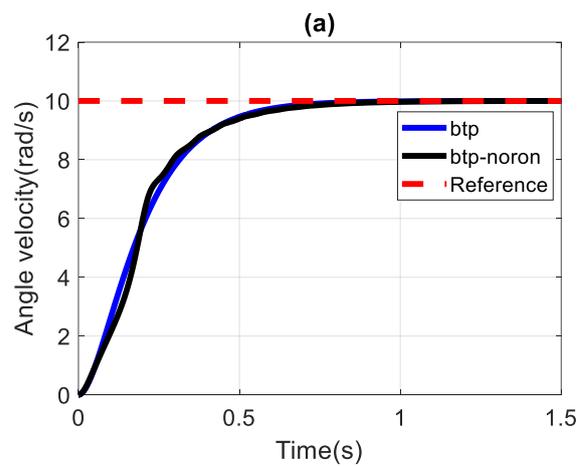


Figure 6. Torque characteristic.



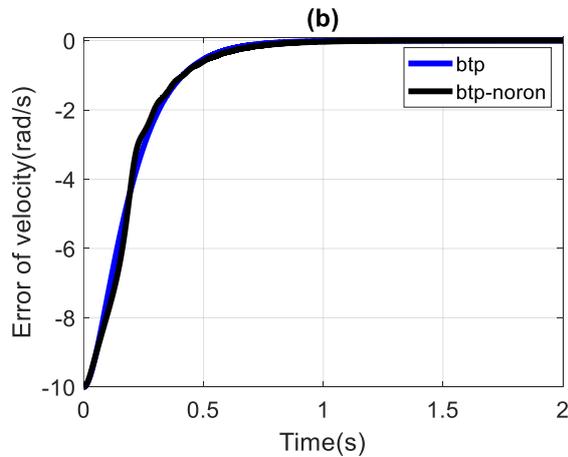


Figure 7. Speed response and error in case of 10 rad/s set point.

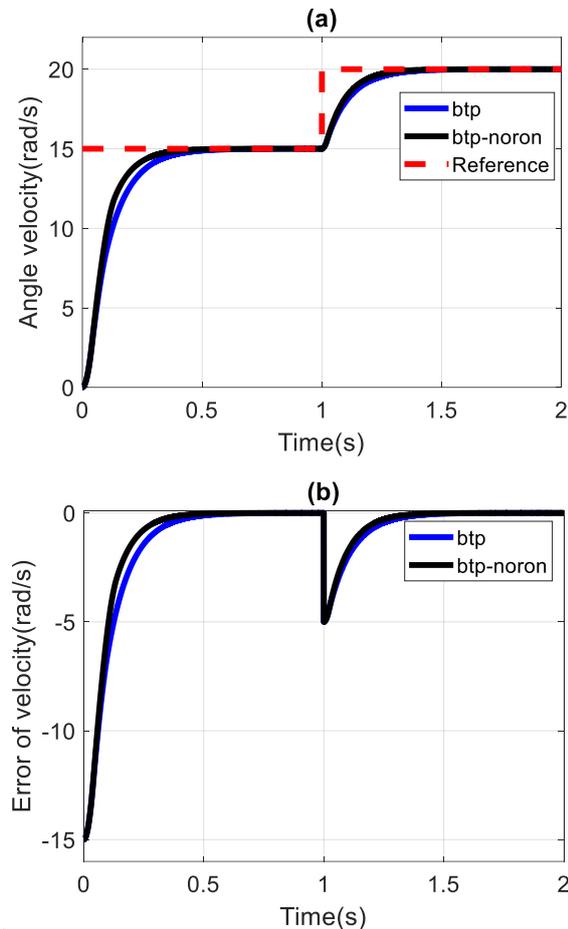


Figure 8. Speed response and the error in case of changing the set point

Simulation results of the SRM control system using the back-stepping controller combined with the magnetic flux estimator by the neural network achieves the desired qualities. The flux approximation error quickly converges to near zero, since the neural network flux estimator has been trained off-line with high accuracy (10^{-5} of SE). When the neural network flux estimator is combined with the back-stepping controller, the control system gives good quality, fast response to set speed with static error almost zero. Torque characteristic (Figure 6) is not good because the logic control of the switches is not optimal in time.

5. Conclusions

This study demonstrates a new approach in SRM control system. In that, a back-stepping controller is combined with ANN based flux estimator. The flux estimator based on artificial neural network has been trained offline and been used to overcome the difficulties in calculating or measuring the motor flux. The simulation results show the effectiveness of back-stepping controller combined with ANN based flux estimator. The ANN could have successfully replaced a mathematic flux models (their coefficients are difficult to determine and depends on the type of SRM, each SRM size,...) as well as experimental values (difficult to measuring) with high accuracy estimation. Besides, the control performance still is guaranteed compared with traditional back-stepping controller. All characteristics of the response satisfy the design criteria such as: steady state static error, settling time, and percentage overshoot.

References

- [1] A. Berdai, A. Belfqih, J. Boukherouaa, F. Mariami, A. Hmidat, V. Vlasenko, V. Titjuk,

- Similarity and Comparison of the Electrodynamics Characteristics of Switched Reluctance Motors SRM with Those of Series DC Motors, *Engineering*, vol. 7, 2015 pp. 36–45.
- [2] A. Nirgude, M. Murali, N. Chaithanya, S. Kulkarni, V. B. Bhole, S. R. Patel, Nonlinear Mathematical Modeling and Simulation of Switched Reluctance Motor, *IEEE International Conference on Power Electronics, Drives and Energy Systems (PEDES)*, 2016, pp. 1–6.
- [3] G. Rigatos, P. Siano, S. Ademi, Nonlinear H-Infinity Control for Switched Reluctance Machines, *Nonlinear Engineering*, vol. 9, no. 1, 2020, pp. 14–27.
- [4] H. L. Huy, P. Brunelle, A Versatile Nonlinear Switched Reluctance Motor Model in Simulink Using Realistic And Analytical Magnetization Characteristics, *31st Annual Conference of IEEE Industrial Electronics Society*, 2005, pp. 1556–1561.
- [5] J. A. Makwana, P. Agarwal, S. P. Srivastava, Modeling and Simulation of Switched Reluctance Motor, *Lect. Notes Electr. Eng.*, 2018, pp. 545–558, <https://doi.org/10.1007/978-981-10-4762-652>.
- [6] K. Deguchi, S. Sumita, Y. Enomoto, Analytical Method Applying a Mathematical Model for Axial-Gap-Switched Reluctance Motor, *Electr. Eng. Japan (English Transl. Denki Gakkai Ronbunshi)*, Vol. 196, 2016, pp. 30–38, <https://doi.org/10.1002/ej.22749>.
- [7] L. Shen, J. Wu, S. Yang, X. Huang, Fast Flux Measurement for Switched Reluctance Motors Excluding Rotor Clamping Devices and Position Sensors, *IEEE Transactions on Instrumentation and Measurement*, Vol. 62, N. 1, 2013, pp. 185–191, <https://doi.org/10.1109/TIM.2012.2212598>.
- [8] L. Zeng, H. Yu, Research on a Novel Rotor Structure Switched Reluctance Motor, *Phys. Procedia*, Vol. 24, 2012, pp. 320–327. <https://doi.org/10.1016/j.phpro.2012.02.048>.
- [9] M. I. Spong, R. Marino, S. M. Peresada, D. G. Taylor, Feedback Linearizing Control of Switched Reluctance Motors, *IEEE Transactions on Automatic Control*, Vol. 32, 1987, pp. 371–379, <https://doi.org/10.1109/TAC.1987.1104616>.
- [10] M. J. Grimble, P. Majecki, *Nonlinear Industrial Control Systems: Optimal Polynomial Systems and State) Space Approach*, Springer-Verlag London, 2020.
- [11] V. T. Nguyen, S. F. Su, N. Wang, W. Sun, Adaptive Finite-Time Neural Network Control for Redundant Parallel Manipulators, *Asian Journal of Control*, Vol. 22, No. 6, 2019, pp. 2534–2542, <https://doi.org/10.1002/asjc.2120>
- [12] V. T. Nguyen, C. Y. Lin, S. F. Su, W. Sun, M. J. Er, Global Finite Time Active Disturbance Rejection Control for Parallel Manipulators With Unknown Bounded Uncertainties, *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, 2020, pp. 1–12, <https://doi.org/10.1109/tsmc.2020.2987056>
- [13] O. Ustun, A Nonlinear Full Model of Switched Reluctance Motor With Artificial Neural Network, *Energy Conversion and Management*, Vol. 50, N. 9, 2009, pp. 2413–2421, <https://doi.org/10.1016/j.enconman.2009.05.025>
- [14] X. Sun, K. Diao, Z. Yang, G. Lei, Y. Guo, J. Zhu, Direct Torque Control Based on a Fast Modeling Method for a Segmented-Rotor Switched Reluctance Motor in HEV Application, *IEEE Journal of Emerging and Selected Topics in Power Electronics*, Vol. 9, No. 1, 2021, pp. 232–241.