

Article

Neuroadaptive Robust Speed Control for PMSM Servo Drives with Rotor Failure

Omar Aguilar-Mejía ^{1,*}, Hertwin Minor-Popocatl ^{1,*} , Prudencio Fidel Pacheco-García ¹ and Ruben Tapia-Olvera ² 

¹ School Engineering, UPAEP University, Puebla 72410, Mexico; omar.aguilar@upaep.mx (O.A.-M.); fidel.pacheco@upaep.mx (P.F.P.-G.)

² Departamento de Energía Eléctrica, Universidad Nacional Autónoma de México, Mexico City 04510, Mexico; rtapia@fi-b.unam.mx

* Correspondence: hertwin.minor@upaep.mx; Tel.: +52-222-229-94-00 (ext. 7431)

Abstract: In this paper, a neuroadaptive robust trajectory tracking controller is utilized to reduce speed ripples of permanent magnet synchronous machine (PMSM) servo drive under the presence of a fracture or fissure in the rotor and external disturbances. The dynamics equations of PMSM servo drive with the presence of a fracture and unknown frictions are described in detail. Due to inherent nonlinearities in PMSM dynamic model, in addition to internal and external disturbances; a traditional PI controller with fixed parameters cannot correctly regulate the PMSM performance under these scenarios. Hence, a neuroadaptive robust controller (NRC) based on a category of on-line trained artificial neural network is used for this purpose to enhance the robustness and adaptive abilities of traditional PI controller. In this paper, the moth-flame optimization algorithm provides the optimal weight parameters of NRC and three PI controllers (off-line) for a PMSM servo drive. The performance of the NRC is evaluated in the presence of a fracture, unknown frictions, and load disturbances, likewise the result outcomes are contrasted with a traditional optimized PID controller and an optimal linear state feedback method.

Keywords: modeling uncertainty; computational optimization method; inertia degradation; fissure mechanism; B-spline neural network; trajectory tracking; on-line learning



Citation: Aguilar-Mejía, O.; Minor-Popocatl, H.; Pacheco-García, P.F.; Tapia-Olvera, R. Neuroadaptive Robust Speed Control for PMSM Servo Drives with Rotor Failure. *Appl. Sci.* **2021**, *11*, 11090. <https://doi.org/10.3390/app112311090>

Academic Editor: Fabio La Foresta

Received: 4 September 2021

Accepted: 3 November 2021

Published: 23 November 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

The Permanent Magnet Synchronous Motors (PMSM), have extraordinary efficiency, excellent performance, wide speed range, fast response, great accuracy compact construction, and higher torque per volume ratio. This motor plays an important protagonist in industrial applications such as chip mount machines, scraper conveyors, marine propulsion, aerospace, hard disk drives, and semiconductor production machines [1–3]. The linear control scheme such as vector control with proportional-integral (PI) loops is the industrial standard control of the PMSM due to its simplicity, excellent performance, and efficiency [2,4]. Nevertheless, the large trouble of the traditional PI controllers is its sensitivity to the parametric variations of the plant (PMSM) [5]. Another issue that is getting interested is faults in electric motors due to vibration problems produced by an imbalance that is caused by degradation in the rotor shaft, which last will origin a fissure in the rotor shaft by the fatigue phenomena [6]. A topic of great interest in the field of engineering is to know and determine the dynamic behavior of a crack in solid materials to contribute to the preservation of the life of mechanical devices [7]. However, designing a precise control scheme for a PMSM is a challenging task due to the complexity of the system caused by nonlinearities, unpredictable parameter variations, external load disturbances and unmolded nonlinear dynamics of the motor.

The set of differential equations that describe the dynamic response of electromechanical systems with the presence of a fissure in the rotor depict a non-linear behavior, causing that traditional tuning techniques for PI controllers present a poor performance, for which

it is necessary to use techniques of tuning based on heuristic algorithms or artificial intelligence techniques [8,9]. Rotor failures have been studied by different groups of researchers. Classical control has been applied to deal with the presence of degradation problems, fault, and broken rotors in electromechanical systems, such as optimal linear quadratic regulator [9,10], sliding mode control [11], feedback linearization controller [12], and fuzzy logic control [13]. In [9] an optimal linear state feedback control scheme was designed using the optimal linear quadratic regulator technique, for a PMSM with rotor failure. A sliding mode control (SMC) for a quad-rotor with a rotor failure has been presented in [11]. The controller has a good performance to a total rotor failure, uncertainties and different weather conditions caused by wind speed. However, by its nature, the SMC has some inherent problems in its operation such as the chattering phenomenon due to the discontinuity in the switching strategy of the control law. In [12] a feedback linearization controller was made to offer better precision control for induction motor drive subject to short circuit faults in the stator winding and some broken bars in the rotor. An observer based on the feedback of states with the capability to compensate for some failures was designed for two coupling PMSM systems [14]. The controller is based on a robust adaptive observer to estimate the PMSM faults. The controller compensates for the faults and allows the motors to track the reference. In [13] a fuzzy adaptive controller was designed to reject the effects caused by electrical faults for PMSM based on dynamic surface technology. Fuzzy logic control (FLC) has been used because it is simple to implement and this controller does not require an accurate mathematical model of uncertain nonlinear systems [15]. However, the gains of an FLC are fixed, require knowledge or experience into membership functions, fuzzy rules, or fuzzy inference system databases. These issues are the main disadvantages of FLC systems design. Some groups of investigators try to overcome this drawback using hybrid PI controllers or novel tuning rules to improve their performance against faults and parameter variations. In [16] an adaptive control structure with neural data processing was designed for two converter-fed DC machines coupled by a flexible connection. The controller is based on a recurrent neural network and is inspired by the Elman model. For the selection of constant coefficients of the controller, the particle swarm optimizer is applied.

In the last twenty years, various control schemes based on B-Spline neural network (BSNN) have been designed because BSNN can compute different control signals due to its adaptability nature through proper online training. The diverse shapes of BSNN have been used to constitute the control schemes for mechatronics and electromechanical systems [17–21].

Recently, in [18] an adaptive PI controller is used to compute on-line the parameters of the regulator structure using a BSNN for a CD motor. The procedure for the BSNN online learning algorithm (OLA) is to calculate the weights and base functions of the neural network at each instant of time while the PI controller is working. Depending on the structure and size of the BSNN, the algorithm computes the different weights of the network considering only the values of the inputs and the desired output, thus obtaining a faster result and at a lower computational cost, in relation to the ANN based on a backpropagation algorithm [20,21].

Seeking to contribute to the diagnosis and control of PMSM subject to temporary degradation due to the effects of a load of use. In this work, we analyze the behavior of PMSM with the presence of fissure in the rotor shaft and unknown frictions; with the analysis and proposed theory. In the work, a detailed analysis of the PMSM modeling with the presence of a fracture in the rotor is carried out, and the procedure to obtain a precise, robust and stable control scheme under different operating conditions, contributing to obtain a more precise mathematical description of the dynamic behavior of the PMSM. Therefore, to solve the aforementioned problem, we design a neuroadaptive robust trajectory tracking controller to reduce speed ripples of PMSM servo drive based on traditional PI controllers with the ability to update their parameters through a BSNN. The principal contribution of the work are: (a) introduction of an adaptive robust neural

trajectory tracking controller for a PMSM with the presence of fissure in the rotor shaft, abrupt changes of load torque disturbance, and unknown frictions; (b) a novel approach is used to simultaneously for adjust multiple parameters of a PMSM control scheme; (c) with the methodology used, the behavior of an adaptive controller with self-tuning capacity is obtained in a practical approach; and (d) with the most optimal search of the PIBSNN parameters, the algorithm learns online, avoiding the use of additional stages for its training. The shapes curve of the activation functions of the BSNN presents several advantages in relation to the radial functions, where the curve changes only in the neighborhood of some control points that have been modified. But the BSNN functions are limited in relation to the maximum and minimum values of the desired rotor speed that can be selected as input signals to the BSNN, then using the gradient descent rules the neuroadaptive controller weights are updated in line. Therefore, the neuroadaptive controller can reject internal and non-parametric disturbances that happen in the PMSM during its operation, warranting a globally stable system against limited disturbances. The controller takes advantage of BSNN to compensate for the abrupt changes of load torque disturbance and unknown frictions.

The remainder of this paper is prepared as follows: In Section 2 an overview of the mathematical model of the PMSM in the dq reference frame with the presence of fissure in the rotor shaft, abrupt changes of load torque disturbance, and unknown frictions are presented. In Section 3, the proposed neuroadaptive controller based on BSNN is set out in detail. To identify the efficiency of the proposed control scheme, numerical simulations are presented in Section 4. Finally, conclusions are detailed in Section 5.

2. Dynamic PMSM Model with Presence of Rotor Fissure

The typical mathematical model of the synchronous machine can be described in the rotor reference frame (dq) coordinates as [1,2]:

$$\frac{di_d}{dt} = \frac{V_d}{L_d} - \frac{r_s}{L_d}i_d + \frac{L_q}{L_d}\omega_r i_q \quad (1)$$

$$\frac{di_q}{dt} = \frac{V_q}{L_q} - \frac{r_s}{L_q}i_q + \frac{L_d}{L_q}\omega_r i_d - \frac{\lambda_m}{L_q}\omega_r \quad (2)$$

where L_d is the d -axes stator inductance and L_q represent the q -axes stator inductance, i_d is the stator current in d -axes, i_q represent the stator current in q -axes, V_d is the stator voltage in d -axes and V_q represent the stator voltage in q -axes, r_s represent the stator resistance per phase, ω_r is the rotor mechanical angular speed and λ_m is the rotor flux. The differential equations that define the dynamics of the PMSM rotor with respect to angular velocity and the angular position (θ) of the rotor are determined as follows [1–3,9,10]:

$$\frac{d\theta}{dt} = \omega_r \quad (3)$$

$$\frac{d\omega_r}{dt} = \frac{1}{J_t} [T_e - T_L - F_f(\omega) - \beta\omega_r] \quad (4)$$

where P is the number of pole pairs, J_t is the moment of inertia taking into account the degradation of the rotor shaft, $F_f(\omega)$ are other disturbances caused by coupling the load torque, β is the viscous friction coefficient and T_L is the load torque. The term T_L can deteriorate the control performance if it is not appropriately estimated. The friction model has been used in [22,23] to describe the compensation of the unknown frictions. In the control law, it should be considered that the mathematical models of friction can be defined as intermittent or continuous events in some instants of time during the operation of the motor, causing the control scheme to calculate non-smooth control signals. Therefore, the friction torque $F_f(\omega)$ can be given by

$$F_f(\omega) = a_1 \tanh(c_1 \omega) + a_2 [\tanh(c_2 \omega) - \tanh(c_3 \omega)] \quad (5)$$

where ω represent the speed rotor; a_1 and a_2 are different friction levels; and c_1 , c_2 , and c_3 represent coefficients to estimated different friction effects and are positive constants. In this work, the following assumptions are made: (a) ω_r , i_d and i_q are available; (b) T_L is unknown, $\dot{T}_L$ is continuously differentiable; (c) $F_f(\omega)$ is continuously differentiable and (d) the desired speed ω_{rd} is constant and $\dot{\omega}_{rd}$, $\ddot{\omega}_{rd}$ are continuously differentiable. The input of dynamic Equation (4) is the motor torque vector T_e which is calculated by

$$T_e = P \left[\frac{3}{2} (L_d - L_q) i_d i_q + \frac{3}{2} \lambda_m i_q \right] \quad (6)$$

Fracture Dynamics in the Rotor Shaft

To estimate how the crack works inside the PMSM rotor, the effects caused by stress generated by the load torque T_l on the rotor will be considered. Then, when there is a crack in the rotor, the fracture opens and closes due to the moment of inertia and the rotational effects of the rotor mass, which over time the rotor failure will behave as a weak fracture. [9,10,24]. The mathematical expression that defines the dynamic behavior of the fissure within the rotor can be described with the Paris equation, as follows [24,25].

$$\frac{da}{dt} = cf \Delta K^n \quad (7)$$

where a is the fissure size, c is the proportional coefficient which depends on the type of rotor material, ΔK is the variation of the stress concentrator, n is the proportional exponent which depends on the type of rotor material and f is the fundamental rotor frequency. The stress intensity factor is defined as [10]:

$$\Delta K = \Delta \tau_{max} \sqrt{\pi a} \quad (8)$$

where $\Delta \tau_{max}$ is the torsional stress variation on the rotor shaft. Considering that the crack will increase its size depending on the degradation dynamics and from Equation (8), the variation of torsional stress $\Delta \tau_{max}$ is defined, taking into account that the fault is in a hollow circular section, for which [26].

$$\Delta \tau_{max} = \frac{16(T_e - T_L)D}{\pi(D^4 - d^4)} \quad (9)$$

where D is the root of rotor shaft diameter and d is the size of the cavity in the rotor shaft due to the fissure, which the smaller diameter of d is defined as a function of the variation in the length of the fissure as [9,10,26]:

$$d = g_a a \quad (10)$$

where, g_a defines the crack respiration ratio used by Mayes and Davis [10,26] represented as [10]

$$g_a = \frac{l + \cos(\omega_r t)}{2} \quad (11)$$

Therefore, the existence of a crack in the PMSM rotor will result in the variation of the rotational inertia coefficient of the rotor and can be represented as follows [9,10,26]

$$J_t = \frac{1}{8} m_s (D^4 - d^4) \quad (12)$$

where m_s is rotor mass.

3. Control Strategy

The PMSM generally works with different speeds, torque conditions, and unpredictable parameter variations. These operating conditions cause speed ripples in the

PMSM rotor. To reduce PMSM servo drive speed ripples under the presence of degradation in rotational inertia, parameter variations and, external disturbances, this work uses an adaptive robust neural trajectory tracking controller. To achieve this goal, it is essential to create a controller with the ability to maintain precision and robustness under the presence of fissure in the rotor shaft, abrupt changes of load torque disturbance, and unknown frictions. To provide these demands, we introduce a neuroadaptive robust speed trajectory tracking controller based on PI B-Spline neural network (PIBSNN), where its weights and PI parameters are adjusted using a traditional optimization algorithm called moth-flame.

3.1. Conventional PI Controller

As shown in Figure 1, the controller technique is based on three cascaded adaptive PI loops one external speed loop and two internal current loops [27]. As conventional in PMSM regulator, the d -axis current desired (i_d^*) is set to zero and the quadratic current desired (i_q^*) is get from the speed external controller.

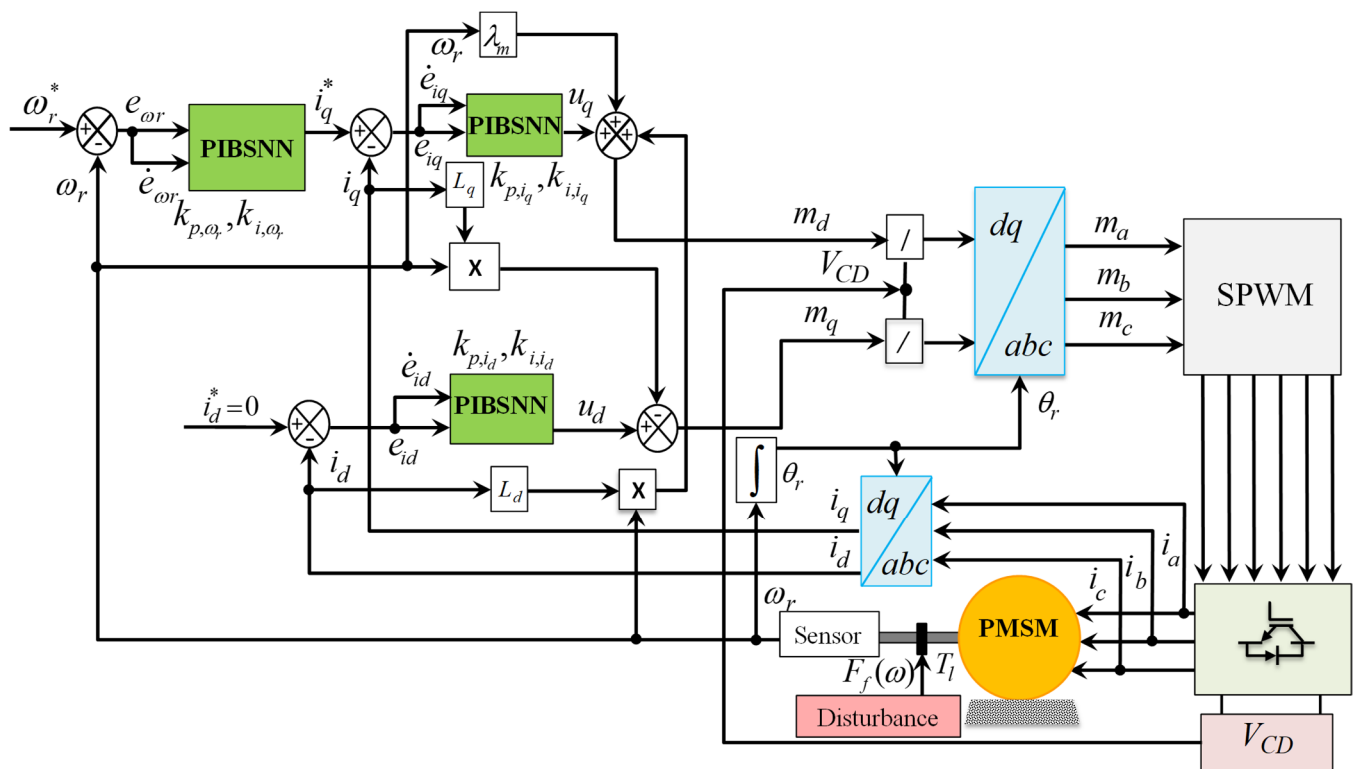


Figure 1. Block diagram of neuroadaptive robust speed control for PMSM servo drives with rotor failure.

The dq -axis entry current controls are computed by two PI decoupling controllers defined as [28]

$$u_d = k_{p,id}(i_d^* - i_d) + k_{i,id} \int_0^t (i_d^*(\tau) - i_d(\tau)) d\tau - i_q \omega_r L_q \quad (13)$$

$$u_q = k_{p,iq}(i_q^* - i_q) + k_{i,iq} \int_0^t (i_q^*(\tau) - i_q(\tau)) d\tau + i_d \omega_r L_d + \lambda_m \omega_r \quad (14)$$

where $k_{p,id}$ and $k_{i,id}$ are proportional and integral parameters of the current on d -axis, respectively; $k_{p,iq}$ and $k_{i,iq}$ are proportional and integral parameters of the current on q -axis, respectively; and

$$\begin{aligned} u_d &= U_d + i_q \omega_r L_q \\ u_q &= U_q - i_d \omega_r L_d - \lambda_m \omega_r \end{aligned} \quad (15)$$

where U_d and U_q are two new auxiliary control variables given by the two inner current control loop PI controllers. To determine the q -axis current reference (for Equation (14)), the rotor speed (ω_r) is used as the feedback signal, which is compared with the desired speed ω_r^* and the speed error (e_{ω_r}) drives a PI controller. The action described above is performed with the external control loop that can be defined as

$$i_q^* = k_{p,\omega_r}(\omega_r^* - \omega_r) + k_{i,\omega_r} \int_0^t (\omega_r^*(\tau) - \omega_r(\tau)) d\tau \quad (16)$$

where k_{p,ω_r} and k_{i,ω_r} are the proportional parameters of rotor speed and the integral parameter of rotor speed, respectively. Thus, k_{p,ω_r} , k_{i,ω_r} , k_{p,i_d} , k_{i,i_d} , k_{p,i_q} and k_{i,i_q} are computed simultaneously online using the BSNN to reject internal or external disturbances and obtain excellent performance in tracking the desired trajectory.

3.2. B-Spline Neural Networks

In the past two decades, adaptive controllers based on BSNN have been used in many industrial, electronic, mechatronics, and electromechanical control systems to regulate output variables, such as parallel kinematic manipulators, electrical power systems, shunt DC motors, UPS inverters, quadrotor, power control of wind turbine and induction motor [16–21]. The typical structure of a BSNN is shown in Figure 2. The main feature of the BSNN is adjusting its weights iteratively on-line in an attempt to reproduce a particular function using a conjugate gradient. The adaptability feature of the neural network gives the ability to learn from previous events, by interconnecting the input data to output, as shown in Figure 2. However, the initial values of weights of the BSNN are often generated randomly or adjusted based on the designer's experience to get excellent performance for a specific controller [16,17,19].

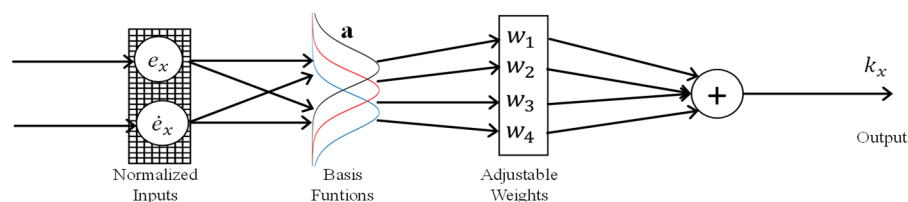


Figure 2. B-Spline neural networks structure.

The BSNN can compute its response on-line, based on a modest structure and a short number of basic mathematical operations that are an important desirable feature for efficiently controlled physical systems. To design the BSNN is a priority to define the size of the input space lattice, which is defined by a set of n th knot vectors (KV). In this work, a KV is used for each one of the three control loops. To define the number of knots, the interval between them, and the value of each one, this can be calculated using an optimization algorithm or through the user's experience knowing the magnitude and the input signals to the neural network. Thus, each control loop has a KV but they must be adjusted in relation to the maximum and minimum values of the reference signals. The KV are distributed in groups of 4 elements and 8-knot points, to form 3 B-spline functions that between them must share one or more knot points. The univariate base function for the B-spline can be defined in the following expression [17]

$$S_K^j(u) = \left(\frac{u - \lambda_{j-K}}{\lambda_{j-1} - \lambda_{j-K}} \right) S_{K-1}^{j-1}(u) + \left(\frac{\lambda_j - u}{\lambda_j - \lambda_{j-K+1}} \right) S_{K-1}^j(u) \quad (17)$$

$$S_1^j(u) = \begin{cases} 1 & \text{if } u \in I_j \\ 0 & \text{for other cases} \end{cases}$$

where u is the input, λ_j is the j_{th} knot point and $I_j = [\lambda_{j-1}, \lambda_j)$ is the j_{th} interval between two-knot points, and K corresponds to the order of the output function. The output of each BSNN can be written as follows [29,30]:

$$\sigma_i = \sum_{m=1}^P a_m w_m = a_i^T w_i \quad \forall_i = 1, 2, 3$$

where a_i corresponds to P -dimensional vector which have the outputs of the basis functions and, w_i corresponds to the weights vector which is updated through the selected instantaneous learning instruction

$$w_i = w_i(t-1) + \frac{\gamma \sigma_i(t)}{\|a_i(t)\|_2^2} a_i(t) \quad \forall_i = 1, 2, 3 \quad (18)$$

where $\|\cdot\|$ is the Euclidean norm and the increment, γ corresponds to the learning rate, $w_i(t-1)$ is the previous weight, and σ_i corresponds to the BSNN output error. The process to carry out the instantaneous training (IT) that uses the neurocontroller based on the BSNN, is used to calculate the value of the weights of the active base functions. The algorithm for the instant learning rule works to minimize the estimated value of performance metric mean square error (PMMSE) function of the output vector, and the values are adjusted using the descending gradient criterion (DGC) [30]. Following the rules defined above, the online training algorithm for the BSNN can be summarized in the next pseudocode (Algorithm 1).

Algorithm 1 B-Spline Neural Network on-line training rules.

```

Input Define:  space lattice with  $n$  knot-vectors
Define:       basis functions ( $K$  order, shape and distribution)
Define:       number of knot-vector
Define:       nodes of hidden layer
Define:       Initial conditions (weights)
Define:       error signal and minimum and maximum values
Define:       threshold error
Load weights,  $K$  order, threshold error, error signal
while  $t <$  simulation time do
    Calculate the input and output value of each layer
    Calculate the errors between target and current value
    if  $ex <$  threshold error
        return  $kx$ 
    else
        Update weights Equation (18)
        Calculate the input and output value of each layer
        return  $kx$ 
end

```

For the initial training stage, the weights and PI parameters are calculated randomly and the off-line training proceeds with DGC in evaluating the PMMSE. The PMMSE is defined as:

$$\text{PMMSE} = \frac{1}{N} \sum_{i=1}^N (x_{rc} - x^*) \quad (19)$$

where N corresponds to the number of input data, x_{rc} corresponds to the computed output, and x^* is the desired output signal. The entire algorithmic process is carried out until the considered performance metric MSE comes to a near possible minimal value. At the time of training off-line, simultaneously the weights of the PIBSNN model are trained for their best values using the classic moth-flame optimization algorithm. In the second stage, the IT rules offer an alternative so that the weights are recurrently updated and reach convergence at finest values. Hence, the PIBSNN training process is continuously carried out online,

while the weight's value is computed occupying only two feedback variables for each PI controller. This can be attained with the BSNN to update k_{p,ω_r} , k_{i,ω_r} , k_{p,i_d} , k_{i,i_d} , k_{p,i_q} and k_{i,i_q} of (13, 14 and 16) at every sampled time. It is important to mention that; if the error signal is less than 3%, the algorithm does not update the corresponding controller weights.

3.3. Overview of Moth-Flame Optimization Algorithm

The moth-flame optimization algorithm (MFOA) was introduced by Mirjalili [31] which mimics the behavior of the moth-flame in the way of traveling during the night. The method way of moving the moth-flame is called transverse orientation. The principal elements in the MFOA are two: the moths and light sources (flame). It is important to highlight that in each iteration of the MFOA, moths, and flames are part of different sequences in the update process. For the above, they can fly in a 1-dimensional, 2-dimensional, 3-dimensional, or hyperdimensional area by changing the position vectors. In the MFOA procedure, we consider that each moth has a position in a solution space of dimension D . An array can express the set of search agents, just as follows:

$$M = \begin{bmatrix} m_{1,1} & \cdots & m_{1,d} \\ \vdots & \ddots & \vdots \\ m_{n,1} & \cdots & m_{n,d} \end{bmatrix} \quad (20)$$

where M is the position matrix of moths, $m_{i,j}$ is the value of j -th parameter of the i -th moth, $j = 1, 2, \dots, d$ and $i = 1, 2, \dots, n$; n is the number of moths and d represents the number of dimensions (variables represented in the problem) in the solution space. The corresponding fitness is represented as:

$$OM = \begin{bmatrix} OM_1 \\ \vdots \\ OM_n \end{bmatrix} \quad (21)$$

where the flame matrix is in the same dimension as the moth matrix. Flame also stores the fitness value accordingly as a number of flames. Both the moth and the flame are considered as part of the solution, the moth in the solution is the search agent and the flame is the best position of the moth. The flames are considered the trigger that indicates the dropped by the moth in the development of the search process and move around this position to proceed to update. Due to this, the moths never lose the best solution. Moths update their position with respect to flame according to Equation (22)

$$M_{i,j} = S(M_i, F_j) \quad (22)$$

where $M_{i,j}$ represents the i -th moth, F_j represents the j -th flame and S is the spiral function. The movement of the moth can be expressed as a spiral logarithmic function represented as:

$$S(M_i, F_j) = D_i e^{bt} \cos(2\pi t) + F_j \quad (23)$$

where b is a constant to define the shape of the logarithmic spiral, t is a random number between $[-1, 1]$ and D_i is the distance between the i -th moth and the j -th flame, can be defined as:

$$D_i = |F_j - M_i| \quad (24)$$

In order to avoid degradation of the most promising solutions, the following equation represents the number of flames in this problem

$$\text{Flame no} = \text{round}\left(N - l \frac{N - l}{T}\right) \quad (25)$$

where l represents = current number of iterations, N represent = maximum number of flames and T represent = maximum number of iterations. The flowchart representation of the MFOA is displayed in Figure 3.

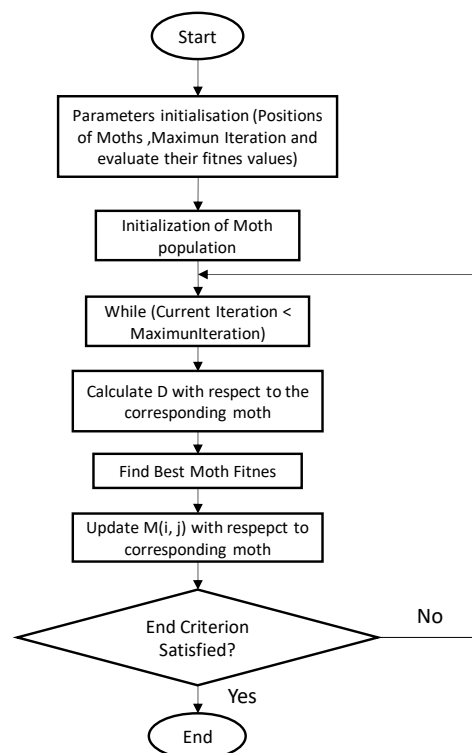


Figure 3. Flowchart of MFOA.

4. Simulation Results

The PIBSNN controller is applied for speed regulation of PMSM. When the PIBSNN is required to generate a specific function in the output vector, it is necessary to train it iteratively offline. The weights of the PIBSNN are adjusted off-line using the MFOA. The complete simulations are implemented MATLAB R2020b and are implemented in a PC with an Intel Core i9 processor with 2.40 GHz speed and 32.00 GB RAM.

The inputs to the PIBSNN are current error in q axis (e_{iq}), current error in d axis (e_{id}), $e_{\omega r}$, and their derivatives ($\dot{e}_{iq}$, $\dot{e}_{id}$ and $\dot{e}_{\omega r}$) respectively. During the progress of training on-line, the neurocontroller adjusts PI controller parameters $k_{p,\omega r}$, $k_{i,\omega r}$, $k_{p,id}$, $k_{i,id}$, $k_{p,iq}$ and $k_{i,iq}$ and regulates the speed of the PMSM rotor efficiently against bounded disturbances. Figure 1 displays the block diagram of PIBSNN controller strategy. This control scheme has fourteen unknown parameters of three adaptive controllers that should be determined by optimization techniques. The design parameters are weights factors ($w_{1,kpx}$, $w_{2,kpx}$, $w_{3,kpx}$, $w_{4,kpx}$, $w_{1,kix}$, $w_{2,kix}$, $w_{3,kix}$, $w_{4,kix}$) and PI parameters ($k_{p,\omega r}$, $k_{i,\omega r}$, $k_{p,id}$, $k_{i,id}$, $k_{p,iq}$ and $k_{i,iq}$). Table 1 presents the parameters and simulation coefficients considered in the simulation of the dynamic PMSM system.

Figure 3 is presented the process of searching for the parameters of the speed control scheme for the nonlinear dynamic system of PMSM. Using the MFOA, the search and test of the operation of the control scheme parameters are carried out. Table 2 displays the parameters used to implement the MFOA. To find the initial conditions of the PI controllers and weights factors on the PIBSNN controller using the MFOA algorithm; a reference speed $\omega_r = 400$ rpm is employed, with a load torque of 4 N.m, $a_1 = 0.2$, $a_2 = 3$, $c_1 = 0.2$, $c_2 = 0.015$, and $c_3 = 0.02$.

Table 1. Parameters for the PMSM model.

Parameters	Value	Units
L_d	6.73	Inductance [mH]
L_q	6.73	Inductance [mH]
r_s	2.6	Resistance [Ω]
P	2	Pairs of poles
J_t	3.5×10^{-5}	Initial rotational inertia [kg/m ²]
m_s	0.1	Rotor mass [kg]
λ_m	0.319	Magnetic flux [Wb]
f	$188.5/2\pi$	Frequency [Hz]
β	1×10^{-4}	Viscous friction coefficient [Nms]
c	10×10^{-11}	Proportional coefficient
n	3	Proportional exponent
D	0.137409	Rotor diameter [m ^{1/2}]
T_L	5	Nominal load torque [Nm]
	120	Nominal voltage [Volts]
	4	Rated current [Amp.]
V_{CD}	250	Direct current voltage bus [Volts]

Table 2. Moth-flame optimization algorithm implementation parameters.

Parameters	Value
Population dimension	50
Maximum of iteration	150
b	1.5
t	0.5

The set of transient responses delivered by the first iteration of the MFOA, which comply with the parameters established in the algorithm, to find the most optimal parameters of the neuroadaptive robust speed control for PMSM servo drives with rotor failure, are shown in Figure 4 in different colors. The best transient response of the rotor speed when the execution of the MFOA ends is shown in Figure 5. The optimal parameters for the neuroadaptive controller found by the MFOA are: $w_{1,kpx} = 390.5$, $w_{2,kpx} = 130.5$, $w_{3,kpx} = 9.1$, $w_{4,kpx} = 362.8$, $w_{1,kix} = 21.45$, $w_{2,kix} = 80.1$, $w_{3,kix} = 127.85$, $w_{4,kix} = 37.1$, $k_{p,\omega_r} = 0.1839$, $k_{i,\omega_r} = 51.71$, $k_{p,id} = 47.08$, $k_{i,id} = 19.2$, $k_{p,iq} = 25.12$, and $k_{i,iq} = 5.11$. In this work, to know the performance and behavior of the neuroadaptive controller, two different contexts are assumed. The first case evaluates when the PMSM has a variable load torque under normal operating conditions. In the second context, to verify the robustness and efficiency of the neuroadaptive controller, a load torque with abrupt variations is added to the rotor shaft and PMSM presents unknown friction in the rotor. The convergence of the selected objective function for optimization by MFO is observed in Figure 6. MFO starts by assigning the positions of the moths randomly within the solution space, as shown in Figure 4. The figure shows that in the first iterations the value of the function is greater than 35, which is in function to the transient response of PMSM as shown in Figure 4; but approximately between iteration 20 and 90, the objective value of the function remains constant at a value of 19.7852 and in the last 50 iterations the MFO algorithm continues updating the positions of the moths and llamas and generating new positions, until the 150 scheduled iterations, but always saving the best solution, shown in Figure 5.

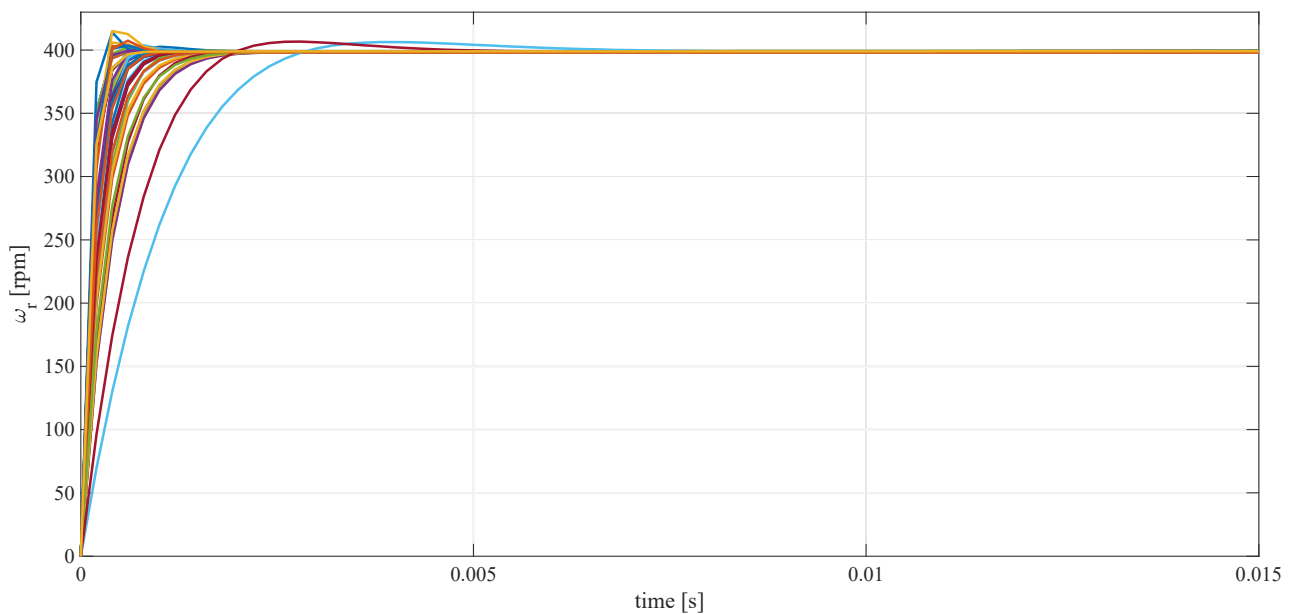


Figure 4. First iteration of the MFO algorithm to calculate the initial parameters of the neurocontroller.

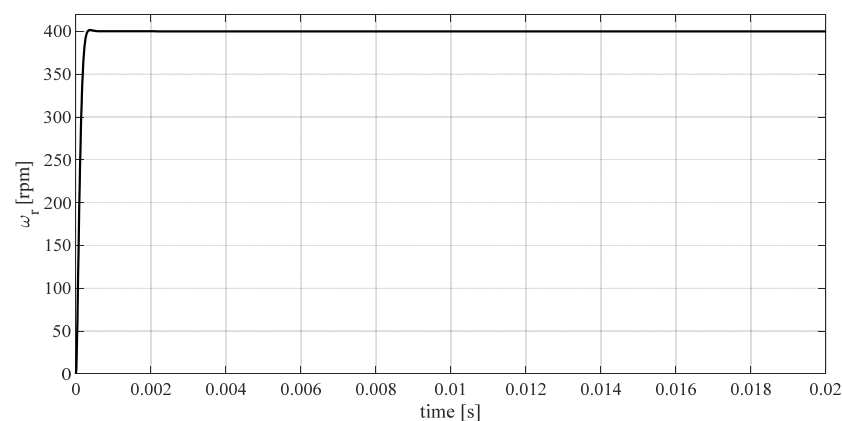


Figure 5. The best transient response of the rotor speed gives as a result the MFOA.

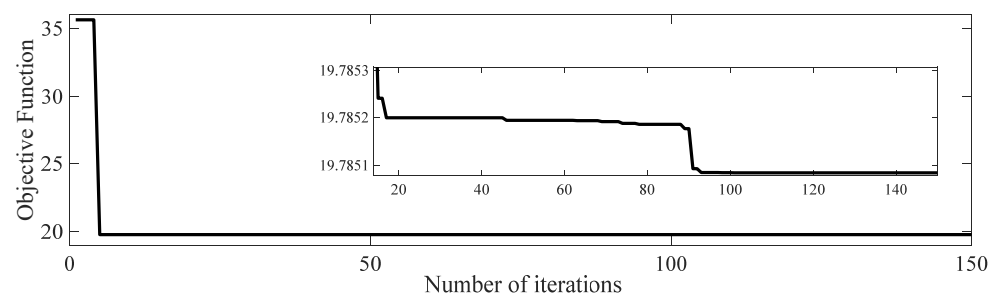


Figure 6. The objective function with respect to iteration numbers of MFO algorithm.

4.1. Simulation Result for Smooth Varying Load Condition

To prove the efficiency and robustness of the neuroadaptive controller, the results of the implementation in software are collated with a traditional optimal PI regulator design technique based on metaheuristic techniques called PI-Op for short [29] and an optimal linear state feedback control scheme (Opt-feed) based on the references [9].

In the first case, a load that varies between 1.7 and 2.1 N.m is coupled to the rotor shaft, at a maximum speed of 400 rpm, considering unknown frictions and a fissure in the

rotor. To generate a smooth path of the desired speed in the rotor between the motor start intervals and the maximum speed, a Bézier polynomial $\mathbb{Y}$ is used, represented as follows:

$$\omega_r^* = \begin{cases} \omega_1 & \text{for } t \leq T_1 \\ \omega_1 + (\omega_2 - \omega_1)\mathbb{Y}(t, T_1, T_2) & \text{for } T_1 < t \leq T_2 \end{cases} \quad (26)$$

where $\omega_1 = 0$ rad/s, $\omega_2 = 400$ rpm, and $\mathbb{Y}$ is the Bézier interpolation polynomial

$$\mathbb{Y} = K^5 [r_1 - r_2 K + r_3 K^2 - r_4 K^3 + \dots - r_6 K^5] \quad (27)$$

$$K = \frac{t + T_1}{T_2 - T_1} \quad (28)$$

with $T_1 = 0$ s, $T_2 = 0.5$ s, $r_1 = 252$, $r_2 = -1050$, $r_3 = 1800$, $r_4 = -1575$, $r_5 = 700$ and $r_6 = -126$.

The efficiency and robustness of the PIBSNN controller for closed-loop tracking of the velocity reference trajectory are established in Figure 7. Simulated waveforms of the dq axis current are plotted in Figure 8a and the simulated waveforms of the speed tracking error e_{ω_r} are displayed in Figure 8b. The performance of the PIBSNN controller is calculated for path following in terms of performance indices: mean squared error (MSE), integral squared error (JISE), and integral absolute error (JIAE). The PIBSNN controller has presented superior transient response as the following error; JIAE and JISE are inferior for the PIBSNN controller. The transient response of the electric currents for closed-loop efficient motor operation exposed to unknown frictions and the presence of a crack in the rotor shaft is suitable, as shown in Figure 8a. The magnitudes of the MSE, JISE, and JIAE, corresponding to the rotor speed error, of the three control schemes, are shown in Table 3. The load profile applied to the PMSM rotor, in case 1 is shown in Figure 9. The MSE (0.5682), JISE (0.2841), and the JIAE (0.2904) are lower for the PIBSNN controller as compared to PI-Op and Opt-feed controller as shown in Table 3.

Table 3. Speed error performance indices for case 1.

Controller	MSE	JISE	JIAE
PIBSNN	0.5682	0.2841	0.2904
PI-Optimal	30.1113	15.0558	1.9064
Optimal linear state feedback	1.8120	0.9060	0.4839

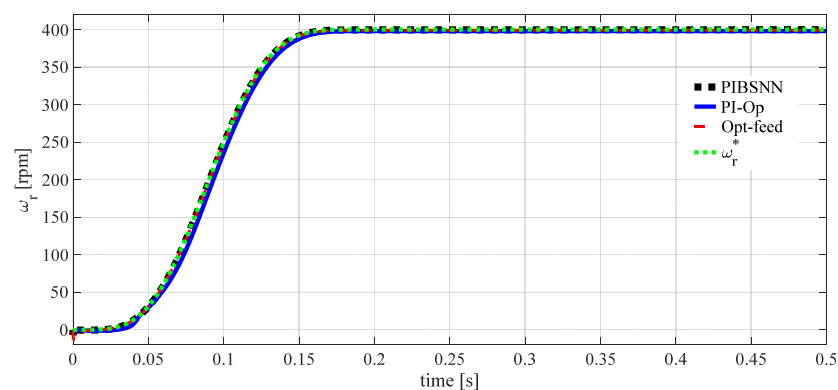


Figure 7. Tracking performance of PMSM drives with a crack in the rotor and unknown frictions.

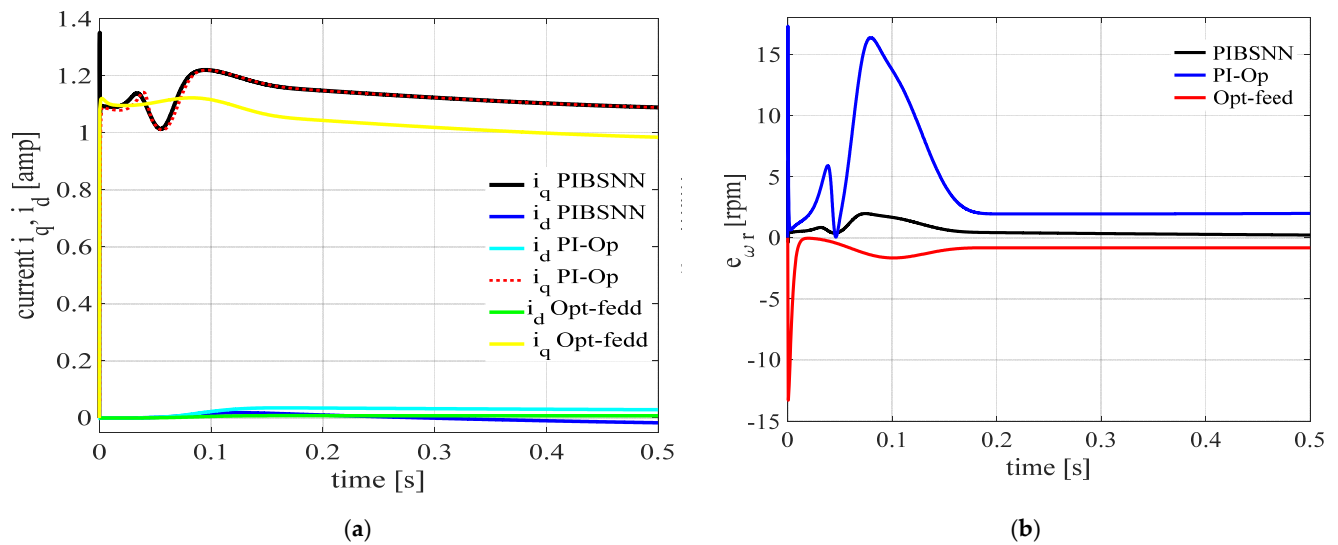


Figure 8. Transient response of the PMSM for case 1 (a) dq axis current and (b) speed tracking error.

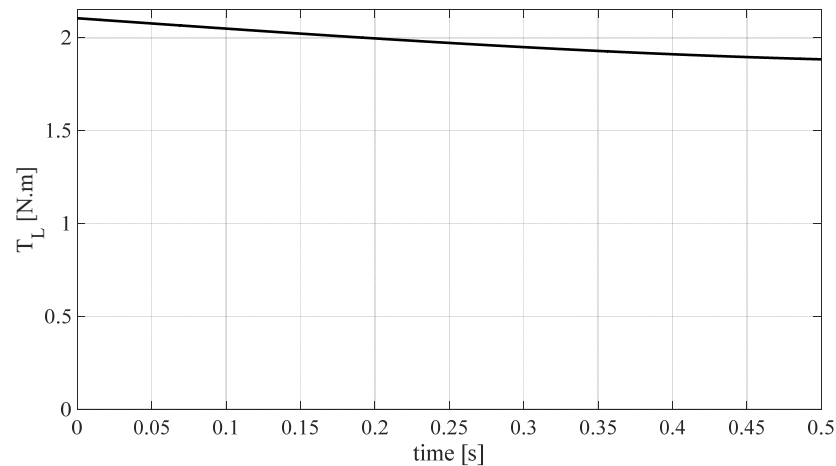


Figure 9. Load torque applied to the PMSM.

4.2. Simulation Result for Abrupt Changes of Load Torque Disturbance

The performance of the PIBSNN, the optimal linear state feedback control scheme and the optimal PI controller are evaluated for speed trajectory tracking application, when the rotor shaft is subjected to abrupt changes of load torque disturbance and unknown frictions. The velocity reference trajectory for a simulation time of 450 s is given by

$$\omega_r^* = \begin{cases} \omega_1 & \text{for } t \leq T_1 \\ \omega_1 \Psi(t, T_1, T_2) & \text{for } T_1 < t \leq T_2 \\ \omega_2 & \text{for } T_2 < t \leq T_3 \\ -\omega_3 - (\omega_2 - \omega_1) \Psi(t, T_3, T_2) & \text{for } T_3 < t \leq T_4 \\ -\omega_3 & \text{for } T_4 < t \leq T_5 \\ -\omega_3 + (\omega_2 - \omega_1) \Psi(t, T_5, T_6) & \text{for } T_5 < t \leq T_6 \\ \omega_4 & \text{for } T_6 < t \leq T_7 \end{cases} \quad (29)$$

where $\omega_1 = 0$ rad/s, $\omega_2 = 800$ rpm, $\omega_3 = -600$ rpm, $\omega_4 = 600$ rpm, $T_1 = 0$ s, $T_2 = 18$, $T_3 = 135$ s, $T_4 = 150$ s, $T_5 = 290$ s, $T_6 = 305$ s, $T_7 = 450$ s and Ψ is a Bézier interpolation polynomial expressed by (25–26) [32]. The PMSM is subjected to variable load torque

described by the sum of sinusoidal functions with different magnitude and frequency as shown in Figure 10 and written as

$$T_l = k_1 + k_2 \sin(1.9t) + k_3 \sin(0.8t), \quad (30)$$

where $k_1 = 2.1$, $k_2 = 0.47$ and $k_3 = 0.42$. In Figure 11 is illustrated the speed behavior angular ω_r and Figure 12 shows the dq axis current of PMSM with fissure and external disturbances in the rotor, for case 2.

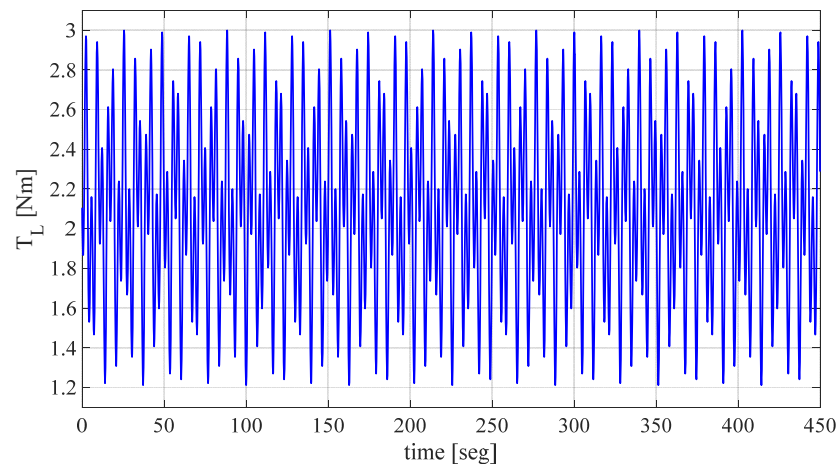


Figure 10. Variable load torque applied to the PMSM.

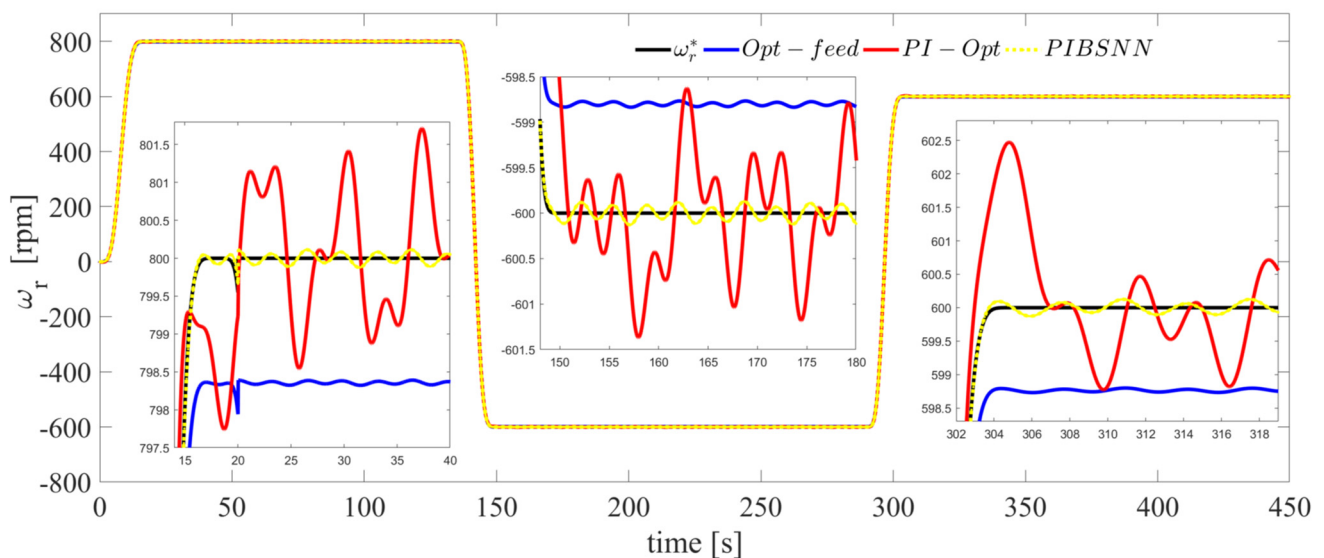


Figure 11. Tracking performance of PMSM drive with presence of fissure in the rotor shaft, abrupt changes of load torque disturbance, and unknown frictions.

Figure 12 shows the dynamic response of the currents delivered by the PI-Op, PIBSNN, and Opt-feed controllers. The tracking error for the PIBSNN, PI optimal controller, and Optimal linear state feedback are shown in Figure 12, when: (a) a torque of load with abrupt changes, (b) it is subjected to unknown vibrations, and (c) there is an internal fracture in the rotor. In Figure 12c it can be seen that the speed error delivered by the control scheme based on a PIBSNN is practically zero because it can compensate for internal and external disturbances mentioned above. The adaptive control parameters computed by the BSNN are shown in Figure 13.

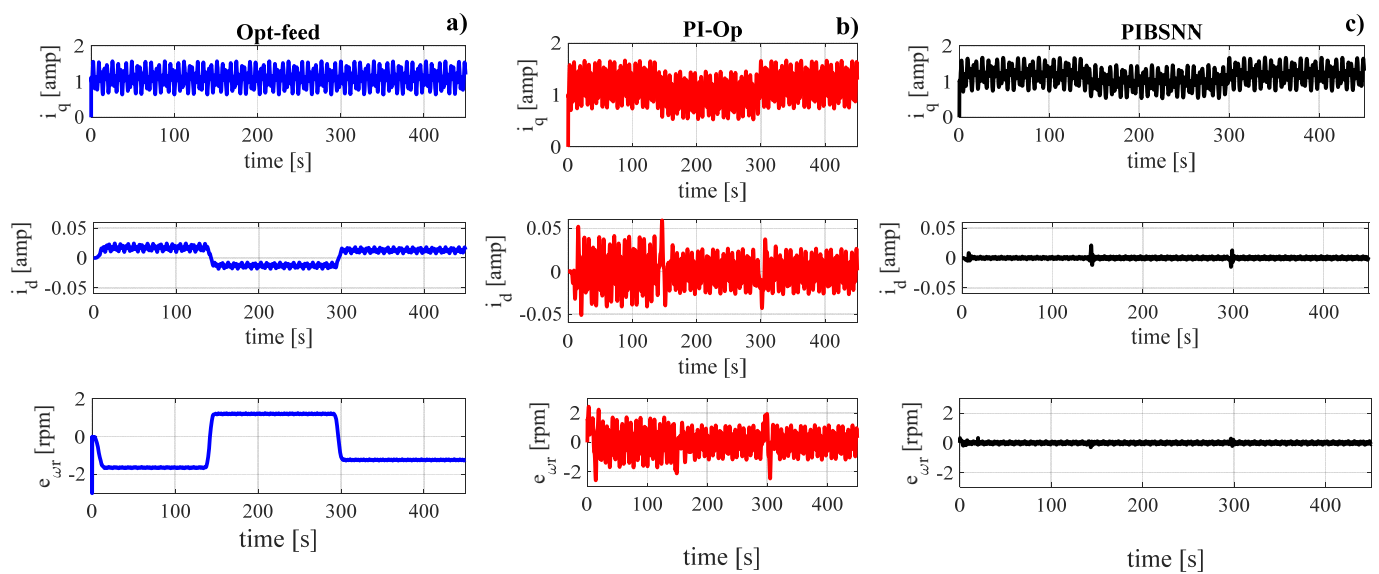


Figure 12. Transient response of currents in the reference frame dq and speed error for the PMSM subjected to abrupt changes of load torque disturbance; (a) Opt.-feed; (b) PI-Op; and (c) PIBSNN.

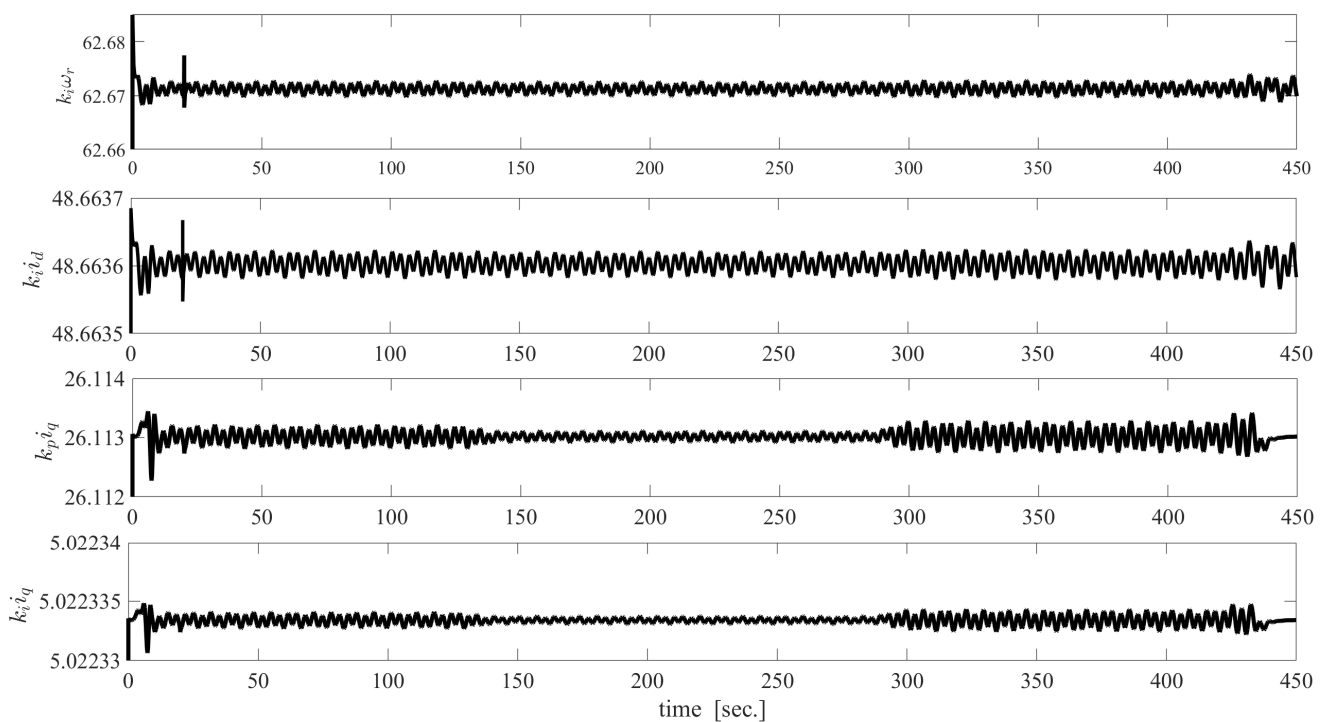


Figure 13. Adaptive control parameters are computed by the B-spline neural network.

Figure 11 shows that the neuroadaptive controller based on a PIBSNN has the robustness and precision necessary to precisely follow the reference speed, even if the motor is subjected to stressful operating conditions produced by external disturbances. Whence, the load is increased or decreased, the PIBSNN controller does not produce any undershoot or overshoot.

The notable performance and robustness of the PIBSNN controller are anew demonstrated in Figure 11 and Table 4. Hence, the PIBSNN depicts an excellent alternative for speed trajectory tracking for the PMSM with fissure in the rotor shaft, subject to abrupt changes of load torque disturbance and unknown frictions. The simulation results for the trajectory tracking control are shown in Figure 11; and MSE, JISE, and JIAE are represented

in Table 4. The MSE (0.0137), JISE (6.1708), and the JIAE (46.2230) are worse for the PIBSNN controller as compared to PI-Opt and Optimal linear state feedback controllers as shown in Table 4. When performing the contrasts of the controllers used in this work, it is easy to see that the adaptive control scheme based on a PIBSNN presents better robustness and an excellent performance against external disturbances, compared with PI-Opt and Optimal linear state feedback controllers.

Table 4. Speed error performance indices for case 2.

Controller	MSE	JISE	JIAE
PIBSNN	0.0137	6.1708	46.2230
PI-Optimal	0.6418	288.7876	289.7450
Optimal linear state feedback	1.7733	797.36	586.1754

The fissure behavior within the PMSM is presented in Figure 14. In Figure 14, the dynamic response of the fissure inside the PMSM rotor is shown, the profile of the fissure shows an exponential growth as a function of the time while the motor is working. From the solution and response of the set of Equations (1)–(12), it is estimated that the crack will continue to grow to cause a degradation in the rotational inertia of the rotor shaft.

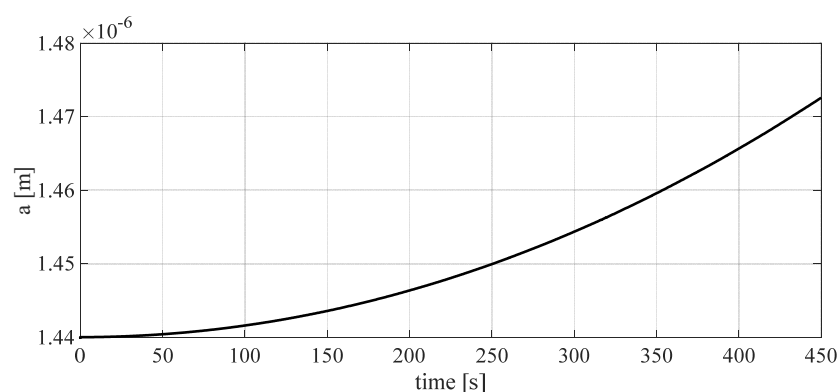


Figure 14. Dynamic behavior of the fissure inside the PMSM rotor.

The PIBSNN controller is an excellent choice for situations where the machine presents a fissure in the rotor shaft and is subject to abrupt changes of load torque disturbance and unknown frictions. It is well known that the PIBSNN controller presents a robust and efficient response because it can reject and compensate for abrupt disturbances of the load coupled to the rotor, fissure in the rotor shaft, and unknown frictions; as compared to PI optimal controller and Optimal linear state feedback controller. Another important aspect to mention is that the profile of the response of the crack inside the rotor is adequate with the expected result because the mathematical model assumes that the crack will continue to grow gradually until the fracture or rupture of the rotor will eventually occur.

5. Conclusions

In this work, neuroadaptive robust speed control for PMSM servo drives with rotor failure was presented. The control scheme of the neuroadaptive controller uses a BSNN because it can adjust the weights of the NN based on the inputs at the low computational effort. Another characteristic of BSNN to reduce computational effort is that it only updates the weights of the neural network when the magnitude of the input signal increases about a predetermined value. In the development of the work, a hybrid control scheme is used where a PI controller and the profits of a BSNN are combined to obtain a neuroadaptive robust speed controller with the ability to reject and compensate for internal and external disturbances. The reference speeds for the motor were based on Bézier interpolation

polynomial to obtain a smooth response from the mechanical and electrical variables of the system.

To verify the performance of the neuroadaptive controller, simulation results confirmed an excellent trajectory tracking robustness against the presence of fissure in the rotor shaft, abrupt changes of load torque disturbance and, unknown frictions. The control scheme was evaluated in the different periods, for two case studies, the transient response performance indices show that the neuroadaptive controller improves its performance by 25% over the other two controllers used. Considering the results presented in this work, it can be concluded that the neuroadaptive controller based on PI and BSNN controllers is an excellent option to regulate the speed of the PMSM, even if it is working in adverse operating conditions such as a fissure in the rotor and abrupt changes in load torque. Furthermore, the performance of the PIBSNN is fairly good according to the comparison with the response of the PI optimal controller and optimal linear state feedback control scheme.

Incoming work, experimental implementations of the neuroadaptive controller to regulate the speed of a PMSM will be developed in a laboratory. Also, we will continue to work and explore the use of other recently published artificial neural networks to design different hybrid controllers with other control techniques.

Author Contributions: Conceptualization, O.A.-M., P.F.P.-G. and R.T.-O.; performed the experiments O.A.-M. and P.F.P.-G. analyzed the data, H.M.-P. and R.T.-O.; writing—original draft preparation, O.A.-M., H.M.-P. and R.T.-O.; writing—review and editing, O.A.-M. and H.M.-P.; funding acquisition, O.A.-M. and H.M.-P. All authors have read and agreed to the published version of the manuscript.

Funding: The authors thank those involved in the project CEMIE-Redes CONACYT-SENER Mexico, B-S-50730 for their support. Aguilar-Mejia, O., Minor-Popocatl, H. and Pacheco-Garcia, P.F. thank the Program for Research of UPAEP for their financial support.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Acknowledgments: Aguilar-Mejia, O. and Minor-Popocatl, H. thank the Program for Research of UPAEP and project CEMIE-Redes CONACYT-SENER Mexico, B-S-50730.

Conflicts of Interest: The authors declare that there is no conflict of interest.

References

1. Pérez, Á.; Saenz-Aguirre, A.; Martín, F.; Vadillo, J. FOC-Droop control strategy for PMSM fed paralleled multi-inverter power systems oriented to aeronautical applications. *Electr. Power Syst. Res.* **2020**, *185*, 106369. [\[CrossRef\]](#)
2. Verrelli, C.V.; Tomei, P. Global stability for the inner and outer PI control actions in non-salient-pole PMSMs. *Automatica* **2020**, *117*, 108988. [\[CrossRef\]](#)
3. Szczepanski, R.; Tarczewski, T.; Grzesiak, L.M. Adaptive state feedback speed controller for PMSM based on Artificial Bee Colony algorithm. *Appl. Soft Comput.* **2019**, *83*, 105644. [\[CrossRef\]](#)
4. Serkies, P.; Szabat, K. Effective damping of the torsional vibrations of the drive system with an elastic joint based on the forced dynamic control algorithms. *J. Vib. Control.* **2019**, *25*, 2225–2236. [\[CrossRef\]](#)
5. Zhang, J.; Wang, S.; Zhou, P.; Zhao, L.; Li, S. Novel prescribed performance-tangent barrier Lyapunov function for neural adaptive control of the chaotic PMSM system by backstepping. *Int. J. Electr. Power Energy Syst.* **2020**, *121*, 105991. [\[CrossRef\]](#)
6. Quiroz, J.C.; Maurin, N.; Rezazadeh, M.; Izadi, M.; Mison, N. Fault detection of broken rotor bar in LS-PMSM using random forests. *Measurement* **2018**, *116*, 273–280. [\[CrossRef\]](#)
7. Bachschmid, N.; Pennacchi, P.; Tanzi, E. *Cracker Rotors: A Survey on Static and Dynamic Behaviour Including Modelling and Diagnosis*; Springer: Berlin/Heidelberg, Germany, 2010; pp. 4–14.
8. Villalobos-Piña, F.; Alvarez-Salas, R. Algoritmo robusto para el diagnóstico de fallas eléctricas en motor de inducción trifásico basado en herramientas espectrales y ondeletas. *Rev. Iberoam. De Automática E Inf. Ind.* **2015**, *12*, 292–303. [\[CrossRef\]](#)
9. Manilla-García, A.; Rivas-Camero, I.; Monroy-Anieva, J. Modelado y Análisis de Sintonización de Velocidad de un MSIP con Presencia de Fisura Mediante Algoritmos Genéticos. *Rev. Iberoam. De Automática E Inf.* **2019**, *16*, 190–199. [\[CrossRef\]](#)
10. Aguilar-Mejía, O.; Manilla-García, A.; Rivas-Camero, I.; Minor-Popocatl, H. Modeling and Speed Tuning of a PMSM with Presence of Fissure Using Dragonfly Algorithm. *Appl. Sci.* **2020**, *10*, 8823. [\[CrossRef\]](#)

11. Hou, Z.; Lu, P.; Tu, Z. Nonsingular terminal sliding mode control for a quadrotor UAV with a total rotor failure. *Aerosp. Sci. Technol.* **2020**, *98*, 105716. [\[CrossRef\]](#)
12. Harzelli, I.; Menacer, A.; Ameid, T. A fault monitoring approach using model-based and neural network techniques applied to input–output feedback linearization control induction motor. *J. Ambient Intell. Hum. Comput.* **2020**, *11*, 2519–2538. [\[CrossRef\]](#)
13. Zhang, G.; Liu, J.; Liu, Z.; Yu, J.; Ma, Y. Adaptive fuzzy discrete-time fault-tolerant control for permanent magnet synchronous motors based on dynamic surface technology. *Neurocomputing* **2020**, *404*, 145–153. [\[CrossRef\]](#)
14. Xiong, H.; Liao, Y.; Chu, X.; Nian, X.; Wang, H. Observer based fault tolerant control for a class of Two-PMSMs systems. *ISA Trans.* **2018**, *80*, 99–110. [\[CrossRef\]](#) [\[PubMed\]](#)
15. Li, H.; Liu, H.; Gao, H.; Shi, P. Reliable Fuzzy Control for Active Suspension Systems with Actuator Delay and Fault. *IEEE Trans. Fuzzy Syst.* **2012**, *20*, 342–357. [\[CrossRef\]](#)
16. Kaminski, M.; Szabat, K. Adaptive Control Structure with Neural Data Processing Applied for Electrical Drive with Elastic Shaft. *Energies* **2021**, *14*, 3389. [\[CrossRef\]](#)
17. Escorcia-Hernández, J.M.; Aguilar-Sierra, H.; Aguilar-Mejía, O.; Chemori, A.; Arroyo-Núñez, J.H. A New Adaptive RISE Feedforward Approach based on Associative Memory Neural Networks for the Control of PKMs. *J. Intell. Robot. Syst.* **2020**, *100*, 827–847. [\[CrossRef\]](#)
18. Tapia-Olvera, R.; Beltran-Carbajal, F.; Aguilar-Mejia, O.; Valderrabano-Gonzalez, A. An Adaptive Speed Control Approach for DC Shunt Motors. *Energies* **2016**, *9*, 961. [\[CrossRef\]](#)
19. Tapia, R.; Aguilar, O.; Minor, H.; Santiago, C. Power System Stabilizer and Secondary Voltage Regulator Tuning for Multi-machine Power Systems. *Electr. Power Compon. Syst.* **2012**, *40*, 1751–1767. [\[CrossRef\]](#)
20. Raya-Armenta, J.M.; Lozano-Garcia, J.M.; Avina-Cervantes, J.G. B-spline neural network for real and reactive power control of a wind turbine. *Electr. Eng.* **2018**, *100*, 2799–2813. [\[CrossRef\]](#)
21. Deng, H.; Srinivasan, D.; Oruganti, R. A B-spline network based neural controller for power electronic applications. *Neurocomputing* **2010**, *73*, 593–601. [\[CrossRef\]](#)
22. Wang, S.; Na, J.; Ren, X. RISE-Based Asymptotic Prescribed Performance Tracking Control of Nonlinear Servo Mechanisms. *IEEE Trans. Syst. Man Cybern. Syst.* **2018**, *48*, 2359–2370. [\[CrossRef\]](#)
23. Yao, J.; Deng, W.; Jiao, Z. RISE-Based Adaptive Control of Hydraulic Systems with Asymptotic Tracking. *IEEE Trans. Autom. Sci. Eng.* **2017**, *14*, 1524–1531. [\[CrossRef\]](#)
24. Arana, J.L.; González, J.J. *Mecánica de la Fractura*; Servicio editorial de la universidad del País Vasco: Bilbao, Spain, 2011; p. 186.
25. Andrade, A.A.; Mosquera, W.A.; Vanegas-Useche, L.V. Modelos de Crecimiento de Grietas por Fatiga. *Entre Cienc. E Ing.* **2015**, *9*, 39–48.
26. Barter, S.; White, P.; Burchill, M. Fatigue Crack Path Manipulation for Crack Growth Rate Measurement. *Eng. Fract. Mech.* **2016**, *167*, 224–238. [\[CrossRef\]](#)
27. Wróbel, K.; Serkies, P.; Szabat, K. Model Predictive Base Direct Speed Control of Induction Motor Drive—Continuous and Finite Set Approaches. *Energies* **2020**, *13*, 1193. [\[CrossRef\]](#)
28. Aguilar-Mejía, O.; Minor-Popocatl, H.; Tapia-Olvera, R. Comparison and Ranking of Metaheuristic Techniques for Optimization of PI Controllers in a Machine Drive System. *Appl. Sci.* **2020**, *10*, 6592. [\[CrossRef\]](#)
29. Brown, M.; Harris, C.J. *Neurofuzzy Adaptive Modelling and Control*; Prentice Hall International: New York, NY, USA, 1994; Chapter 8.
30. Boukens, M.; Boukabou, A.; Chadli, M. Robust adaptive neural network-based trajectory tracking control approach for nonholonomic electrically driven mobile robots. *Robot. Auton. Syst.* **2017**, *92*, 30–40. [\[CrossRef\]](#)
31. Mirjalili, S. Moth-flame optimization algorithm: A novel nature-inspired heuristic paradigm. *Knowl. Based Syst.* **2015**, *89*, 228–249. [\[CrossRef\]](#)
32. Beltrán-Carbajal, F.; Silva-Navarro, G. Arias-Montiel, Active Vibration Control in a Jeffcott-like Rotor with Variable Speed Using an Electromechanical Suspension. *Rev. Iberoam. De Automática E Inf. Ind.* **2014**, *11*, 295–303. [\[CrossRef\]](#)