Fault Diagnosis in the Brushless Direct Current Drive Using Hybrid Machine Learning Models

K. V. S. H. Gayatri Sarman[1†], **Tenneti Madhu**[2], and **A. Mallikarjuna Prasad**[3], Non-members

ABSTRACT

The brushless direct current (BLDC) motor drive is gaining popularity due to its excellent controllability and high efficiency. This paper introduces a fault diagnosis method for open circuit (OC) and short circuit (SC) BLDC motor drives using a hybrid classifier with hybrid optimization. Features such as current, voltage, speed, and torque are considered as the training data. The features are extracted by discrete wavelet transform (DWT) and then employed to train the classifiers to distinguish between fault types and values of response parameters using the support vector machine and Naive Bayes classifier (SVM-NB). To further improve the performance of the system, hybrid chaotic particle swarm optimization (CPSO) algorithms and teaching-learning-based optimization (TLBO) are used. This method is capable of detecting and recognizing the type of faults in the BLDC motor. The developed approach is implemented on the MATLAB/SIMULINK for OC, SC, and no-fault conditions. These hybrid algorithms provide better performance compared to existing approaches with respect to sensitivity, accuracy, and specificity. This improved model achieves about 98.8% accuracy.

Keywords: Brushless Direct Current Moter, BLDC Motor, Open Circuit, Short Circuit, Support Vector Machine, Naive Bayes, Teaching-Learning-Based Optimization, Chaotic Particle Swarm Optimization

1. INTRODUCTION

A motor is an electrical machine that converts electrical energy into mechanical energy. Among the many types of motors used in recent studies, the brushless direct current (BLDC) has the advantage of not needing a mechanical commutator [1]. It offers high dynamic response, efficient controllability, and better efficiency and can be used in various applications such as those involving electric vehicles, the chemical industry, and aerospace systems [2]. In uncertain situations, it may have a predominant effect on system safety, reliability, and product efficiency. Therefore fault diagnosis and localization are needed to monitor the operation of the device.

Various faults may occur in BLDC motors due to electromechanical defects [3]. The BLDC has a better torque to speed ratio, high efficiency, noiseless operation, electromagnetic interference in comparison to the existing DC motor, brushed DC motor, AC induction motor, and conventional DC motor. However, the BLDC can fail because of high current, high load, and demagnetization [4]. The identification of faults in electrical machines and power systems is a growing concern for academics as well as industry [5]. Exposed to manufacturing defects, the BLDC's electrics can render it susceptible to incipient faults or progressive decay, which, if left undetected, can result in system failure. Most BLDC electrical derivative failures disrupt processes, reduce output, and can harm the machinery associated with them. Often a minor failure may result in hours of work stoppage in continuous processing industries and workplaces, where the equipment mounted requires a stable and safe BLDC electrical derivative operation [6].

There are many ways to fix the problem, such as preventive and corrective maintenance, retention of spare motors, protective devices, etc. In some industries, very costly scheduled maintenance has been carried out to avoid unexpected engine failures. Therefore, there is increasing demand to reduce repair costs and avoid unplanned downtime for electric motors and electrical drive systems involving BLDC [7]. Early detection of faults or proper identification and recognition of faults requires a regular maintenance schedule to minimize failure and downtime and increase the overall performance of BLDC electrical drives [8].

There are three major stages in the fault diagnosis process, the first being feature extraction, which is a method of dimensionality reduction in primary data. Many feature extraction methods have previously been used as fast Fourier transform (FFT), Hilbert–Huang transform (HHT), wavelet transform (WT), continuous wavelet transform (CWT), and short-time Fourier transform (STFT) [9–13]. However, all approaches have some disadvantages. For instance, the FFT algorithm does not work well in non-stationary and non-linear signals, WT is ineffective for characterizing the non-linear signals, and STFT has the disadvantage of a fixed-width windowing function [14].

Manuscript received on November 29, 2021; revised on April 9, 2022; accepted on June 1, 2022. This paper was recommended by Associate Editor Yuttana Kumsuwan.

[1]The author is with the Vishnu Institute of Technology, Bhimavaram, Andhra Pradesh, India.

[2]The author is with the Sasi Institute of Technology, Tadepalligudem, Andhra Pradesh, India.

[3]The author is with the Jawaharlal Nehru Technological University, Kakinada, Andhra Pradesh, India.

[†]Corresponding author: gayatrisarman@gmail.com

Digital Object Identifier: 10.37936/ecti-eec.2022203.247517

In the second stage, classification is carried out by methods like the *k*-nearest neighbors (KNN), Naive Bayes (NB), decision trees (DT), support vector machine (SVM), and artificial neural networks (ANNs) [15]. When large datasets are used, NB becomes ineffective, while the minor variation in DT information can cause a major difference in the DT structure. The KNN classification should compute the distance and arrange all the training data for every prediction. In addition, KNN is a lazy learner [16]. Therefore hybrid classification can be employed to improve the efficiency of the system. Recently, many metaheuristic approaches have mainly exploited the fault diagnosis process to improve efficiency, such as the artificial bee colony (ABC), teaching-learning-based optimization (TLBO), particle swarm optimization (PSO) algorithm, and genetic algorithm (GA). These methods suffer from slow searching, slow convergence, and balancing between exploration and exploitation [17–20].

1.1 Motivation

BLDC motors provide various advantages over other DC motors like reduced noise, high efficiency, reliability, high torque to weight ratio, a prolonged lifetime by reducing electromagnetic interference (EMI) and eliminating commutator erosion. However, these motors often undergo open and short circuit faults. Since BLDC motors are utilized in various applications, unexpected faults such as open and short circuits will occur. When a fault occurs, the motor can be operated without breakdown, and it is therefore important to maintain the motor to ensure it works continuously. Many fault detection approaches have been introduced, and it is important to be able to detect faults while they are still developing.

Several feature extraction, classification, and optimization techniques have previously been used to diagnose faults. To improve accuracy, sensitivity, specificity, and efficiency, a hybrid classification technique is used. Metaheuristic approaches suffer from slow convergence and are trapped by local optima; hence, a hybrid optimization algorithm is employed in this study. Therefore, both OC and SC faults can be diagnosed using this developed approach.

Contributions of the paper are:

- A new method for fault diagnosis in open and short circuits is proposed. This method comprises feature extraction, classification, and an optimal solution using hybrid optimization methods.
- Features like current, voltage, speed, and torque are considered as the training data.
- These features are extracted using discrete wavelet transform and then used to train the classifiers to distinguish between fault types.
- The classification of the system is improved by the hybrid support vector machine and Naive Bayes classifier (SVM-NB) to achieve better classification accuracy.
- To further improve the performance of the system, the hybrid algorithm chaotic particle swarm optimization (CPSO) and teaching-learning-based optimization (TLBO) and (CPSO- TLBO) are used.

The remaining structure of the research paper is arranged as follows: the recent related research works are presented in Section 2. Section 3 explains the developed scheme. Section 4 provides a discussion of the implemented results, and Section 5 presents the overall conclusion.

2. RELATED WORKS

Zandi and Poshtan [21] introduced a method for the demagnetization and bearing fault detection of BLDC motors on the basis of hall sensors. Initially, the effect of bearing faults on the speed of the BLDC motor were studied, and the features then extracted from the speed signal using DWT and CWT. Next, the demagnetization of the fault effects on the hall sensors outputs were studied, and the condition of the rotor categorized into healthy or demagnetized using the kurtosis index of the duty cycle signal.

Aker *et al.* [22] presented a model for the detection and identification of faults occurring in the shunt compensated static synchronous compensator (STATCOM) transmission line. Here, feature extraction was performed by DWT and classified using NB and faults studied, such as line to line (LL), double line to ground (LLG), line to ground (LG), and three-phase (LLLG). The results proved that the developed classification approach outperforms with respect to certain measures.

Hosseini *et al.* [23] proposed a fault detection technique for the stator inter-turn fault (SITF) in BLDCM. Initially, the DWT was applied to five types of faults, two of which were constant and varied over time. The third fault was time variable, initiated by the stable phase of stator time and current. The fourth fault was detected by an algorithm checking load torque in the motor, while the fifth fault involved a faulty condition in two phases of the stator rather than one. Following the execution of each case, DWT was applied, and the results reveal the performance of the proposed method.

Shifat and Hur [24] proposed a fault detection and identification method for use in BLDC motors using ANN. The motor current signature detected the faults and calculated the frequency and magnitude of the third harmonic. The faults in the hidden state were extracted from the current and vibration signals. The vibration signal was decomposed using complete ensemble empirical mode decomposition (CEEMD), while the PCA and monotonicity score were used for fault classification. The proposed decomposition method was found to generate better accuracy of 98%.

Lu and Wang [1] demonstrated a BLDC motor rotating phase using variable-speed conditions. Here, the noisy current signal was eliminated by a zero-phase filter, the current phase computed by Hilbert transform (HT), and the signal alignment performed to improve the angle estimation accuracy. The sinusoid similarity was developed to evaluate the phase error of phase estimation

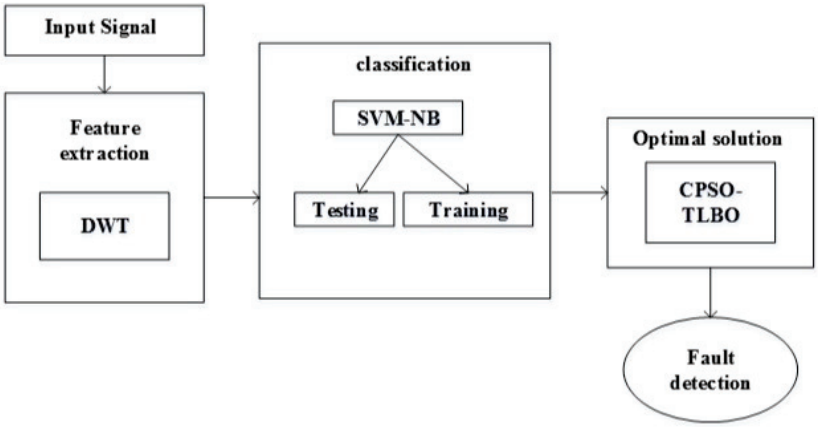

***Fig. 1**: Architecture of the proposed methodology.*

and guide the optimal filter. The accurate curve angle was exploited for resampling to perform fault detection on the basis of the order analysis. This methodology ensures an accurate and simple solution for estimating the BLDC motor rotating phase and is used in various applications.

2.1 Problem Statement

The hybrid models DWT and CWT [21] provide robustness while reducing system costs. They work in both stationary and non-stationary BLDC operations but suffer in information redundancy and orientation selectivity. The method [22] offers better results in classification with and without the STATCOM model, but the precision factor decreases in small datasets. DWT [23] removes faults caused by internal elements and is also applied to the steady and transient states of the current but produces high noise. It also has the ability to find the source of the effect.

CEEMD-ANN [24] achieves high accuracy and three health state classifications, but the number and amplitude of the added noise need to be selected in advance. In certain cases, CEEMD requires empirical post-processing, while HT [1] ensures a non-invasive and accurate solution for estimating the rotating phase of the BLDC motor. However, in some cases, the system is affected by noise and, therefore, the accuracy of the system. Therefore, the proposed method is developed to diagnose OC and SC faults by hybrid classification and optimization, thereby achieving better accuracy, sensitivity, and specificity.

3. PROPOSED METHODOLOGY

This section explains the SC and OC faults in the BLDC motor and feature extraction by DWT classification using the SVM-NB. The appropriate fitness solution using the hybrid method CPSO-TLBO is then subsequently presented.

Fig. 1 illustrates the framework of the proposed methodology. In this method, the input signal is given to the feature extraction. Here, the feature extraction is performed by DWT, which is more efficient. The extracted features are given to classification. Here, the classification is performed by the SVM-NB classifier to improve the accuracy. Furthermore, the performance of the system is improved by hybrid optimization techniques called hybrid CPSO-TLBO.

3.1 OC and SC Faults in the BLDC Motor

Fault identification is considered to be important. As a result, the two mechanisms of fault detection and recognition are sometimes referred to as fault diagnosis. In this section, two faults, namely OC and SC in the BLDC motor, are analyzed. The BLDC motor is capable of generating the necessary angular momentum and torque. The mathematical model of the BLDC is very similar to that of the traditional DC motor, with the notable exception that the phases in the BLDC output have an impact on the inductive and resistive characteristics of this module structure. The stator winding in the coupled circuit is given in Eq. (1).

$$\begin{bmatrix} V_{as}-V_n \\ V_{bs}-V_n \\ V_{cs}-V_n \end{bmatrix} = \begin{bmatrix} R_s & 0 & 0 \\ 0 & R_s & 0 \\ 0 & 0 & R_s \end{bmatrix} \begin{bmatrix} I_a \\ I_b \\ I_c \end{bmatrix} + \frac{d}{dt} \begin{bmatrix} L_{aa} & L_{ab} & L_{ac} \\ L_{ba} & L_{bb} & L_{bc} \\ L_{ca} & L_{cb} & L_{cc} \end{bmatrix} \begin{bmatrix} I_a \\ I_b \\ I_c \end{bmatrix} + \begin{bmatrix} E_a \\ E_b \\ E_c \end{bmatrix} \tag{1}$$

where R_s is the stator resistance phase, the stator current phases of a, b, c are I_a, I_b, I_c, the electromotive forces of each phase are represented as E_a, E_b, E_c, phase to phase voltage is denoted as V_{as}, V_{bs}, V_{cs}, and V_n is the voltage at the neutral node.

Fig. 2 represents the electrical equivalent circuit of the BLDC motor drive. If the three phases are symmetric, self, and mutual inductances, and the variation in rotor reluctance negligible along with the rotor position, Eq. (1) can be written in Eq. (2) as

$$\begin{bmatrix} V_a \\ V_b \\ V_c \end{bmatrix} = \begin{bmatrix} R_s & 0 & 0 \\ 0 & R_s & 0 \\ 0 & 0 & R_s \end{bmatrix} \begin{bmatrix} I_a \\ I_b \\ I_c \end{bmatrix} + \frac{d}{dt} \begin{bmatrix} L & M & M \\ M & L & M \\ M & M & L \end{bmatrix} \begin{bmatrix} I_a \\ I_b \\ I_c \end{bmatrix} + \begin{bmatrix} E_a \\ E_b \\ E_c \end{bmatrix} \tag{2}$$

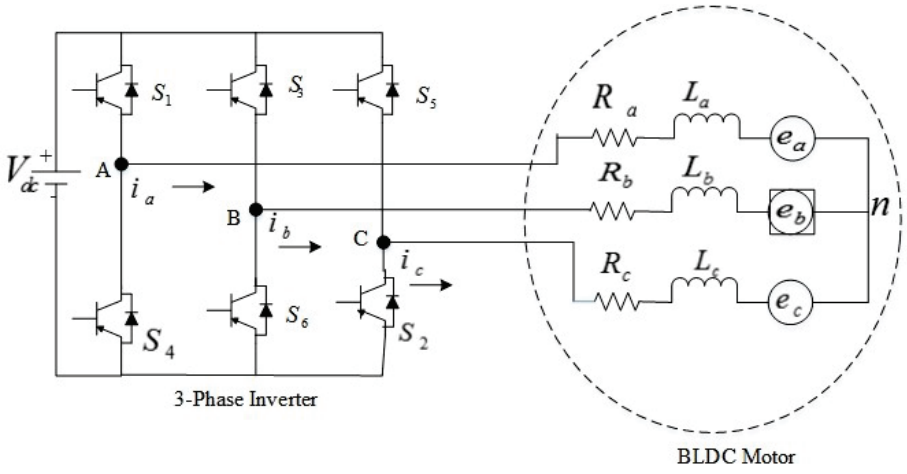

***Fig. 2**: Electrical equivalent circuit of the BLDC.*

where L and M denote the phase inductance and mutual stator winding, respectively.

Simplifying Eq. (2) and the electromagnetic torque Eq. (3) is written as

$$T_e = \frac{E_a I_a + E_b I_b + E_c I_c}{\omega} \tag{3}$$

and the induced emfs are

$$\begin{aligned} E_a &= f_a(\theta)\lambda\omega \\ E_b &= f_b(\theta)\lambda\omega \\ E_c &= f_c(\theta)\lambda\omega \end{aligned} \tag{4}$$

where the electromotive forces of each phase are represented as E_a, E_b, E_c.

The electromechanical torque for the motor is given in Eq. (5).

$$J\frac{dw}{dt} + B\omega = T_e - T_l \tag{5}$$

where J is the moment of inertia, B is the coefficient of friction, and T_l is the load torque.

The position and speed of the rotor are given in Eq. (6) and p is the number of poles in the motor,

$$\frac{d\theta}{dt} = \left(\frac{p}{2}\right)\omega \tag{6}$$

By substituting Eq. (4) in Eq. (2), Eqs. (7) to (9) are obtained,

$$\frac{d}{dt}I_a = \frac{v_a}{L-M} - \frac{R_s}{L-M}I_a - \frac{f_a(\theta)\lambda\omega}{L-M} \tag{7}$$

$$\frac{d}{dt}I_b = \frac{v_b}{L-M} - \frac{R_s}{L-M}I_b - \frac{f_b(\theta)\lambda\omega}{L-M} \tag{8}$$

$$\frac{d}{dt}I_c = \frac{v_c}{L-M} - \frac{R_s}{L-M}I_c - \frac{f_c(\theta)\lambda\omega}{L-M} \tag{9}$$

Dividing Eq. (5) by J and substituting T_e from Eq. (3) and E_a, E_b, E_c values from Eq. (4), we get Eq. (10),

$$\frac{dw}{dt} = -\frac{B}{J}\omega + \frac{f_a(\theta)\lambda I_a}{J} + \frac{f_b(\theta)\lambda I_b}{J} + \frac{f_c(\theta)\lambda I_c}{J} \tag{10}$$

Using Eqs. (7), (8), (9), (10), and (6) from the state space model,

$$\frac{d}{dt}x(t) = Ax(t) + Bu(t) \tag{11}$$

choosing states I_a, I_b, I_c, ω, and θ,

$$\frac{d}{dt}\begin{bmatrix} I_a \\ I_b \\ I_c \\ \omega \\ \theta \end{bmatrix} = \begin{bmatrix} \frac{-R_s}{L-M} & 0 & 0 & \frac{a(\theta)\lambda}{J} & 0 \\ 0 & \frac{-R_s}{L-M} & 0 & \frac{b(\theta)\lambda}{J} & 0 \\ 0 & 0 & \frac{-R_s}{L-M} & \frac{c(\theta)\lambda}{J} & 0 \\ \frac{f_a(\theta)}{J} & \frac{f_b(\theta)}{J} & \frac{f_c(\theta)}{J} & \frac{-B}{J} & 0 \\ 0 & 0 & 0 & \frac{p}{2} & 0 \end{bmatrix}\begin{bmatrix} I_a \\ I_b \\ I_c \\ \omega \\ \theta \end{bmatrix} + \begin{bmatrix} \frac{1}{L-M} & 0 & 0 & 0 \\ 0 & \frac{1}{L-M} & 0 & 0 \\ 0 & 0 & \frac{1}{L-M} & 0 \\ 0 & 0 & 0 & \frac{-1}{J} \\ 0 & 0 & 0 & 0 \end{bmatrix}\begin{bmatrix} V_a \\ V_b \\ V_c \end{bmatrix} \tag{12}$$

From Eq. (12) the values of A and B are obtained.

3.1.1 Open circuit faults

In BLDC drive phases, three types of OC faults occur: single phase A, two phase A and C, and two phase A and

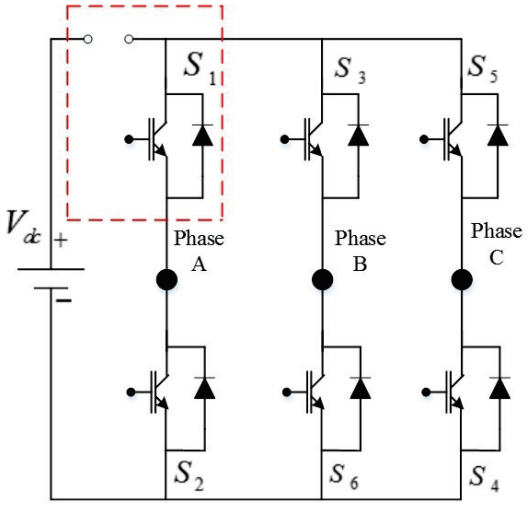

***Fig. 3**: BLDC-OC fault design.*

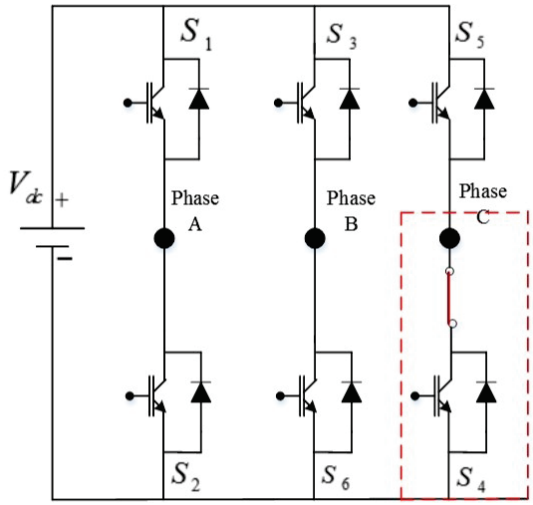

***Fig. 4**: BLDC-SC fault design.*

B. These faults can affect the drive, as the motor faults change from normal speed. The OC faults are mainly caused by a broken conductor, improper installation, high temperature, etc. In unloaded conditions, various types of multi and single faults will occur. In the BLDC drive, two or three phases will be affected if upper or lower switch failure occurs. Fig. 3 shows the BLDC-OC fault design.

3.1.2 Short circuit faults

SC faults occur in three-phase BLDC. The SC is caused by mechanical damage, insulation, overheating, installation, etc. The motor terminal will be affected by faulty impedance. The BLDC output power drive is then affected because of SC faults. Faults such as single and three-phase are classified as SC faults. Three-phase faults occur in inverter switches and single phase faults in the BLDC drive. If the fault occurs among neutral and terminal points, then it is known as a single phase fault. The SC voltage is lower in comparison to the normal voltage level. Fig. 4 presents the BLDC-SC fault design.

3.2 Feature Extraction by DWT

DWT [25] is a more efficient technique for solving data redundancy problems and eliminating unnecessary data. Providing proper and adequate data for signal analysis, DWT significantly minimizes computation time. In addition, it also has an advantage compared to high and low pass filters in being able to decompose the time-domain signal into large scales for different frequency bands, varying the sampling rate in every step. The DWT is created in binary form to choose the position and scale based on the power of 2. The transformation output is achieved by replacing $a = 2^j$ and $b = k \cdot 2^j$, Eq. (13) can be written as

$$wc(a,b) = \frac{1}{\sqrt{|a|}} \int_{-\infty}^{\infty} x(t)\psi\left(\frac{t-b}{a}\right)dt \qquad (13)$$

where $wc(a,b)$ is the wavelet coefficient with scale a and time b, $1/\sqrt{|a|}$ is the normalization, ψ is the wavelet function and $(t-b)/a$ is the shift in time.

The high and low frequency components are approximate and detail coefficients, where A and D are the low and high frequency bands of the signal. This approach is demonstrated by a series of a low and high pass filters as given in Eq. (14)

$$x(t) = A_j + \sum_{j<J} D_j \qquad (14)$$

where A_j and D_j are the low and high frequency bands of the signal. The high frequency components are used to analyze the signal during the transient state. Furthermore, the DWT exhibits aspects of data like self-similarity and breakdown point discontinuities and is more effective than other signal processing approaches.

3.3 Classification by the Hybrid Support Vector Machine and Naive Bayes Classifier

The SVM approach [26] works on the basis of supervised learning. It has become popular because of its ability to solve nonlinear classification problems using the kernel function. The algorithm has a hyperplane which divides the data into classes. The SVM classifier utilizes the hyperplane to isolate categories. Support vectors are used to describe the margin of the hyperplane by learning the examples nearest to it. As a result, the approach also maximizes the margin.

The optimal hyperplane created by the SVM is given in Eq. (15)

$$(w \cdot x) + b = 0 \qquad (15)$$

where, w is the weight vector, b the bias vector, and x the input feature vector. (x_i, y_i) is used to define the sample, then $x_i \in R^d$ and $y_i \in \{-1, +1\}$, where d is the vector dimension if the sample set is divided by the hyperplane, in which case Eq. (16) is then satisfied.

$$\left[y_i(w \cdot x_i) + b\right] - 1 \geq 0, \quad i = 1, 2, \ldots, n \tag{16}$$

To increase the distance among these two sample sets, Eq. (17) must be minimized,

$$\phi(\omega) = \frac{1}{2}||w||^2 = \frac{1}{2}(w \cdot w) \tag{17}$$

The optimization problem is solved and given in Eq. (18),

$$\sum_{i=1}^{n} y_i a_i^0 (x x_i) + b = 0 \tag{18}$$

where a_i^0 is the Lagrange multiplier. If the training set is not dividable, the relaxation and penalty parameters $\in_i > 0$ and C are evaluated. The optimal function is then converted into Eq. (19), and the constraints given in Eq. (20),

$$\phi(\omega) = \frac{1}{2}(w \cdot w) + C \sum_{i=1}^{n} \in_i \tag{19}$$

$$\left[y_i(w \cdot x_i) + b\right] - 1 + \in_i \geq 0, \quad i = 1, 2, \ldots, n \tag{20}$$

SVM takes the sign of $\{y_i\}$ to differentiate samples from various sectors as given in Eq. (21),

$$I(x) = \text{sgn}\left(\sum_{i=1}^{n} y_i a_i^0 \left(\phi(x) \cdot \phi(x_i)\right) + b\right) z \tag{21}$$

where $\phi(x)$, x, and y are the high and low dimensional vectors. Furthermore, x and y are related by the kernel function $K(x, y)$ and no high computational complexity exists.

The Bayesian naive classifer [27] uses Bayes' theorem, and is appropriate when the input dimensionality is high. Hence the classification problem can be written as in Eq. (22),

$$C^* = \arg\max p(c|d) \tag{22}$$

The fundamental probability model can be defined as an independent feature. NB exploits the Bayes' rule as given in Eq. (23)

$$p(c|d) = \frac{p(c)p(d|c)}{p(d)} \tag{23}$$

where $p(c)$ is the probability that a random sample falls into category c. $p(d)$ does not involve selecting C^*. To estimate the factor $p(d|c)$, NB decomposes it by considering f_i to be independent to the d class in Eq. (24), where m is the total number of features and f_i is the feature vector.

$$p_{NB}(c|d) := \frac{p(c)\left(\prod_{i=1}^{m} p(f_i|c)^{n_i(d)}\right)}{p(d)} \tag{24}$$

After feature extraction by DWT, the extracted features are tested and trained using NB and SVM. The drawback with Naive Bayes is the assumption of independence in the feature vectors extracted from the training sets. In addition, SVM can determine a perfect hyperplane to split the training samples into two stages. The aim of SVM is to maximize the distance between the boundaries of the two stages. The assumption of independence in the accuracy and recall rate affects the classification based on NB.

To overcome this problem, SVM employs a trimming approach to eliminate samples that are split into the wrong categories by NB. Accordingly, the dependence of the feature vector is minimized, and the independence between the training samples increased. This hybrid classification architecture is evaluated by combining the merits of the NB (its simplicity, fast classification speed, and the small amount of data required for estimating the parameters) with the high classification accuracy of SVM. By integrating these two methods into the diagnosis of OC and SC faults in LCDC motors, this hybrid combination can be effectively used for classification with low computational complexity.

3.4 CPSO Algorithm

To determine the optimal parameters and maximize device efficiency, the damping controller's parameters are optimized using CPSO [28]. The aim of this algorithm is to increase the damping ratio of all modes to achieve a defined value. The updated solution of the velocity is given in Eq. (25)

$$\begin{aligned} V_{id}(K+1) = {} & \omega V_{id}(k) + C_1 r_1 \left[pbest_{id}(k) - x_{id}(k)\right] \\ & + C_2 r_2 \left[gbest_{id}(k) - x_{id}(k)\right] \end{aligned} \tag{25}$$

The updated solution of the position is given in Eq. (26)

$$x_{id}(k+1) = x_{id}(k) + V_{id}(K+1) \tag{26}$$

where $x_{id}(k)$ and V_{id} are the position and velocity of the i^{th} for the d^{th} dimension at k^{th} iteration, ω is an inertia weight, r_1 and r_2 are random numbers ranging from 0 and 1. C_1 and C_2 are positive coefficients between 0 and 2, namely $C_1 + C_2 \leq 4$ and the i^{th} particle's best position while the entire swarm for the d^{th} dimension is denoted by $pbest_{id}(k)$ and $gbest_{id}(k)$, respectively. PSO suffers from slow searching, slow convergence, and lack of a balance between exploration and exploitation.

A logistic map is given in Eq. (27)

$$Z_t = \alpha Z_{t-1}(1 - Z_{t-1}) \;\; t = 1, 2, \ldots \tag{27}$$

where Z_t is the t^{th} chaotic number, Z_0 is a random number, while the interval is (0,1) and $Z_0 \notin \{0.0, 0.25, 0.5, 0.75, 1.0\}$, set $\alpha = 4$ and is a control parameter.

$$\begin{aligned} X_{id} &= a x_{(i-1)d}(1)(1 - x_{(i-1)d}(1)) \\ V_{id} &= a x_{(i-1)d}(1)(1 - x_{(i-1)d}(1)) \end{aligned} \tag{28}$$

when $1 \leq i \leq M$; $1 \leq d \leq D$, where X_{id} and V_{id} are initial velocities and positions for producing first particle velocities and positions at the first iteration, $V_{0j}(1)x_{0j}(1) \notin \{0.0, 0.25, 0.5, 0.75, 1.0\}$. Likewise, the logistic map utilizes the two independent random values r_1 and r_2 to improve the diversity of the CPSO solution space, as given in Eq. (29).

$$\begin{pmatrix} \omega_k & a\omega_{k-1} & (1-\omega_{k-1}) \\ r_{1k} & ar_{1,k-1} & (1-r_{1,k-1}) \\ r_{2k} & ar_{2,k-1} & (1-r_{2,k-1}) \end{pmatrix} \tag{29}$$

when $1 \leq k \leq n$, where ω_k is the k^{th} chaotic inertia weight, r_{1k} and r_{2k} are the two k^{th} independent chaotic random sequences ω_0, r_{1k}, $r_{2k} \notin \{0.0, 0.25, 0.5, 0.75, 1.0\}$ and random numbers in the range of (0,1). The velocity update equation of the CPSO algorithm can then be rewritten as Eq. (30).

$$\begin{aligned} V_{id}(K+1) = {} & \omega V_{id}(k) + C_1 r_{1,k} \left[pbest_{id}(k) - x_{id}(k)\right] \\ & + C_2 r_{2,k} \left[gbest_{id}(k) - x_{id}(k)\right] \end{aligned} \tag{30}$$

3.5 TLBO Algorithm

TLBO [29] is based on the learner's quality and evolved from the mean value of the class. The learning method in this optimization comprises two parts: teacher and learner. The aim of this method is to ascertain the impact of the teacher on the learner's output in a class and the interaction among learners. TLBO needs parameters such as the size of the population and generation numbers. The algorithm procedure is given below:

Teacher Part: The teacher's goal is to raise the class mean to the appropriate level. However, to a certain degree, this can be accomplished through practice, depending on the class's capacity for learning. Teacher part is formulated in Eq. (31).

$$X_{i,new}^{(t)} = X_i^{(t)} + rand(T^{(t)} - TF \times M^{(t)}) \tag{31}$$

The learning factor TF is randomly found with the probability of 1 and 2. $T^{(t)}$ is the best fitness value with the student selected as teacher of the present iteration t and $M^{(t)}$ is the mean value. $X_i^{(t)}$ and $X_{i,new}^{(t)}$ are the old and new student positions. The individual's position is replaced by a new solution when the new solution is better than the old.

Learner Part: The learners develop their skills in two distinct ways: teacher interaction and learning by themselves. The formulation of the learner part is given as follows. Individuals n_1 and n_2 are chosen for the i^{th} learner in the class, in such a way that when $n_1 \neq n_2 \neq i$, the new equation can be written as in Eq. (32).

$$X_{i,new}^{(t)} = \tag{32}$$

$$\begin{cases} X_i^{(t)} + rand(X_{n1}^{(t)}) - (X_{n2}^{(t)}) & \text{if } X_{n1}^{(t)} \text{ is better than } X_{n2}^{(t)} \\ X_i^{(t)} + rand(X_{n2}^{(t)}) - (X_{n1}^{(t)}) & \text{else} \end{cases} \tag{33}$$

The new solution can be accepted if it has a better function value.

3.6 The Procedure of the CPSO-TLBO

The aim of the hybrid CPSO and TLBO methods is to overcome the individual drawbacks of CPSO and TLBO and utilize their advantages to improve the search quality, ultimately achieving a better solution. The major disadvantage of the PSO is that the parameter tuning is trapped by local optima, while TLBO has low-speed convergence. In the searching process, the whole population is combined, and for the next iteration, the best solutions are chosen as the initial population. The parameter is initialized using Eq. (33), where is the random number range from [0,1]

$$P = X_{\min} + rand(X_{\max} - X_{\min}) \tag{34}$$

Each individual is evaluated and updated using Eq. (32), and the better solution chosen.

Here, the CPSO with the best solution is selected as *gbest*, and the related population situation is then updated. The new population for each solution is compared to the relevant *pbest*. Considering TLBO, in this process, the population with the less favorable solution is chosen as the class teacher. If the final condition of CPSO-TLBO is satisfied, then the process is terminated; otherwise, the algorithms will again evaluate the individual and repeat the process until the criteria are met.

3.7 Time Complexity

The computational complexity of the proposed and existing methods is based on parameters such as the number of iterations N, population size P, and accuracy A. It can be written in big O notation as $O(P \times N \times A)$.

4. RESULTS AND DISCUSSION

All the processes were carried out using a system with an Intel Core i5 CPU, 3.0 GHz speed, and 8 GB RAM. The developed approach was implemented on the MATLAB/SIMULINK. Initially, the current, voltage, speed, and torque under OC and SC faults and no-fault conditions are determined. Measures such as accuracy, sensitivity, and specificity of classifiers like (SVM-TLBO, SVM-CPSO, SVM-TLBO-CPSO, NB-TLBO, NB-CPSO, NB-TLBO-CPSO, SVM+NB-TLBO, SVM+NB-CPSO) are compared with the proposed SVM-NB-CPSO-TLBO method. Finally, the step response parameters are calculated, such as rise time, settling time, overshoot, undershoot, and peak time.

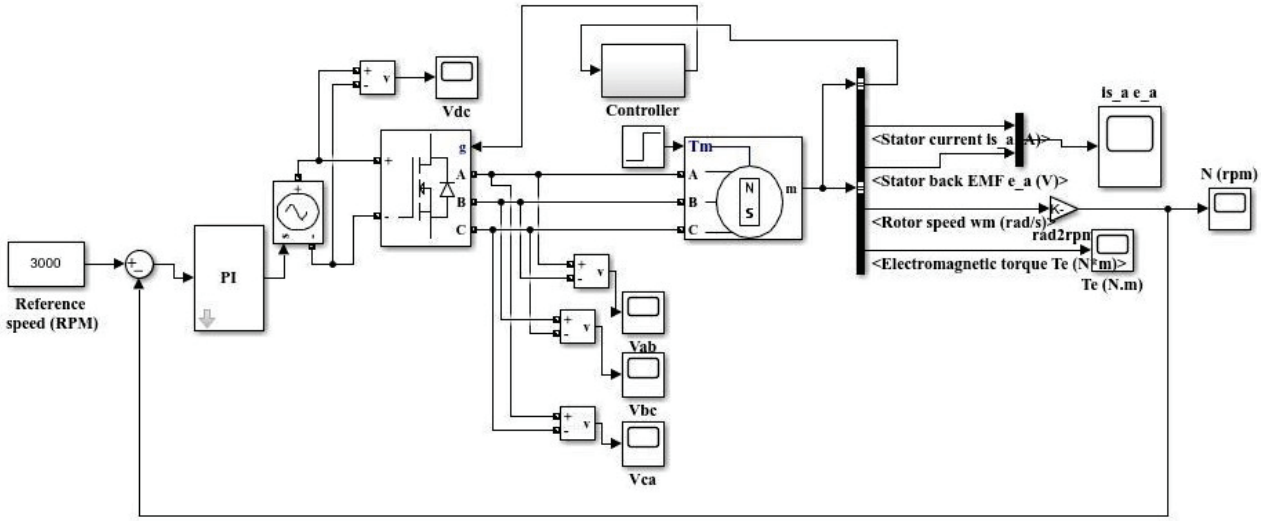

Fig. 5: Simulink model of the BLDC motor under no-fault conditions.

4.1 Performance Measures

The number of accurately classified data instances compared to the total number of data instances is given in Eq. (34), where FP, TP, FN, and TN refer to false positive, true positive, false negative, and true negative, respectively.

$$\text{Accuracy} = \frac{TP + TN}{TP + FP + TN + FN} \tag{35}$$

Sensitivity is defined as the ratio of positives identified by the classifier, as given in Eq. (35).

$$\text{Sensitivity} = \frac{TP}{TP + FN} \tag{36}$$

Specificity is defined as the ratio of negatives identified by the classifier, as given in Eq. (36).

$$\text{Specificity} = \frac{TN}{FP + TN} \tag{37}$$

Rise time is the time needed for the response to rise from 0 to 100% of its final value, as given in Eq. (37), where ω_d is a damped frequency.

$$\text{rise time} = \frac{\pi - \theta}{\omega_d} \tag{38}$$

Peak time is the time needed for the response to reach the peak value for the first time, as in Eq. (38).

$$\text{peak time} = \frac{\pi}{\omega_d} \tag{39}$$

Overshoot can be calculated using the formula given in Eq. (38).

$$\text{overshoot} = \left(e^{-\left(\frac{\delta\pi}{\sqrt{1-\delta^2}} \right)} \right) \times 100 \tag{40}$$

4.2 Simulation Setup of the BLDC Motor

The simulation setup of the BLDC motor under OC, SC, and no-fault conditions is explained in this section.

Fig. 5 shows the simultation setup of the BLDC drive model under no-fault conditions. A 1 kW three-phase BLDC motor has been designed to test the operation. A speed of about 3000 is applied to the PI device to find the speed, voltage, torque, and current under three faults conditions.

4.3 Performance of the BLDC Motor under No-Fault Conditions

In this section, the speed, current, torque, and voltage performance of the BLDC motor under no-fault conditions is provided.

Fig. 6 shows the BLDC no-fault conditions; (a) current, (b) voltage, (c) speed, and (d) torque performance by varying the time. A 40 A current and 450 V voltage are maintained in BLDC no-fault conditions. When the time increases, the torque and speed are maintained within a particular range because no faults occur. The best output EMF voltage from the BLDC motor is provided by CPSO-TLBO, thus reducing fault instances while maintaining a healthy motor.

4.4 Performance of the BLDC Motor under SC Fault Conditions

In this section, the speed, current, torque, and voltage performance of the BLDC motor under short circuit fault conditions are presented.

Fig. 7 shows the BLDC performance with SC fault conditions; (a) current, (b) voltage, (c) speed, and (d) torque. The current, voltage, speed, and torque will be calculated by varying the time for BLDC-SC fault conditions. Here, in the BLDC design, a short circuit fault occurs at switch S_4, as shown in Fig. 4. Thus, the current and voltage are not within the exact range at the time

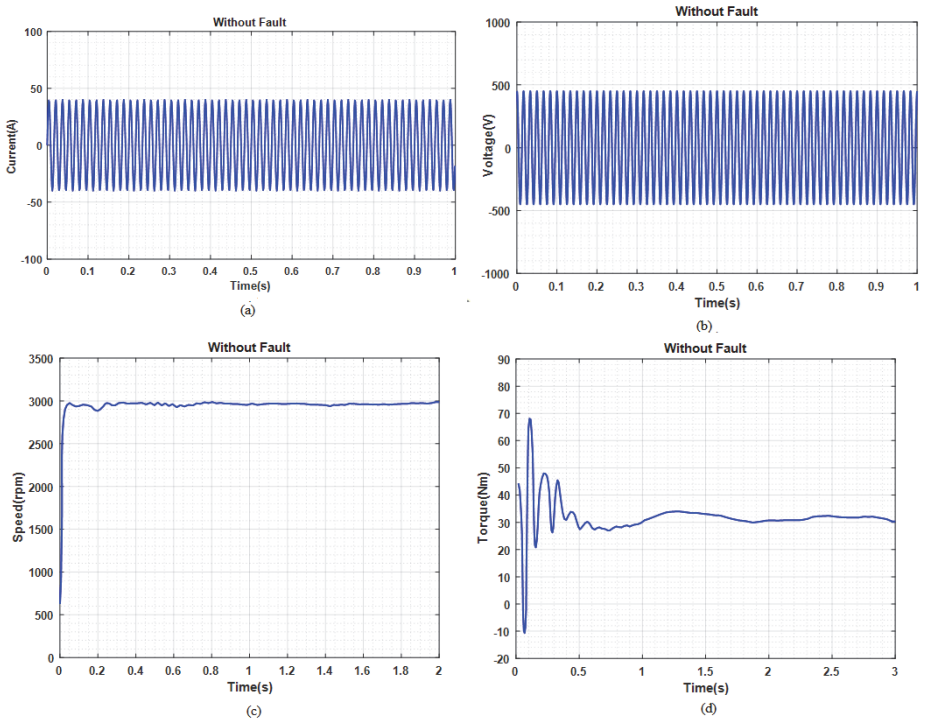

***Fig. 6**: BLDC no-fault conditions; (a) current, (b) voltage, (c) speed, and (d) torque.*

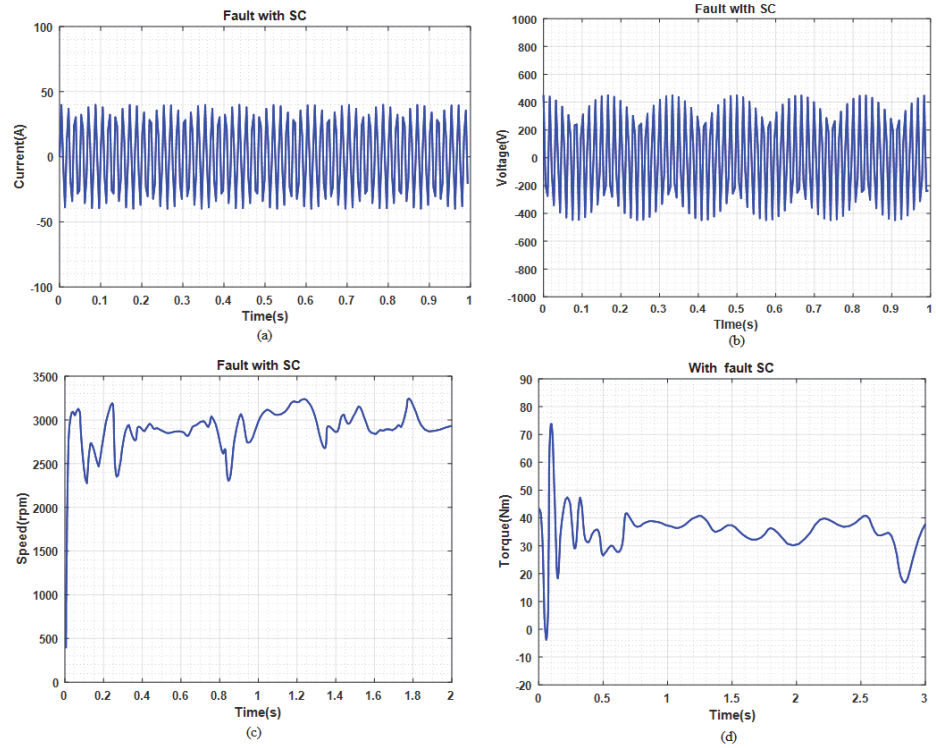

***Fig. 7**: BLDC-SC fault conditions; (a) current, (b) voltage, (c) speed, and (d) torque.*

interval because of the SC fault. The extracted frequency from the DWT relates to the fault frequency range. Also, the speed and torque of the motor will not be in a constant range when the SC fault occurs in the BLDC drive.

4.5 Performance of the BLDC Motor under OC Fault Conditions

In this section, the performance of the speed, current, torque, and voltage of the BLDC motor under OC fault conditions is presented.

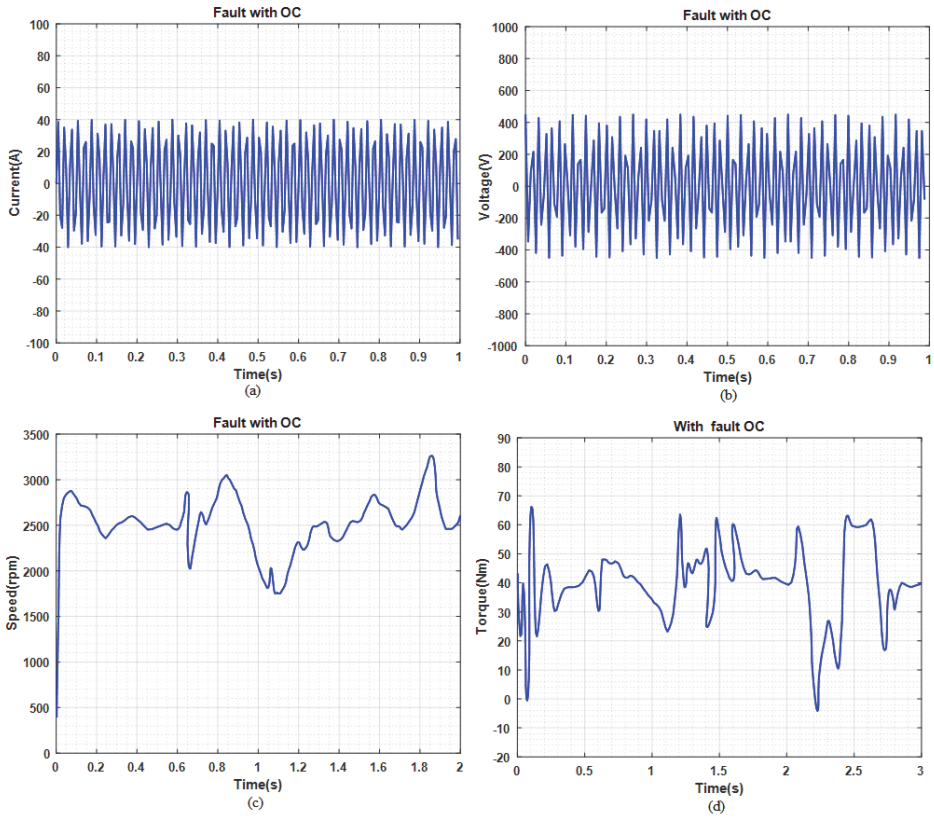

***Fig. 8**: BLDC-OC fault conditions; (a) current, (b) voltage, (c) speed, and (d) torque.*

***Table 1**: Performance of no-fault, SC, and OC faults with respect to speed.*

Fault type	Rise time	Settling time	Overshoot time	Undershoot time	Peak time
No-Fault	0.0062	5.0000	454.1863	1.2216×10^{3}	0.0254
SC Fault	0.0052	4.7995	1.4244×10^{3}	2.3754×10^{-6}	3.100
OC Fault	0.1580	0.7450	9.2146×10^{8}	1.6289×10^{3}	0.5000

***Table 2**: Performance of no-fault, SC, and OC faults with respect to torque.*

Fault type	Rise time	Settling time	Overshoot time	Undershoot time	Peak time
No-Fault	0.0085	5.0000	72.9446	33.4184	0.0243
SC Fault	0	2.6980	1.5133×10^{23}	3.9161×10^{25}	2.6000
OC Fault	2.9957×10^{-8}	0.7450	1.0405×10^{7}	2.4416×10^{15}	0.5000

Fig. 8 shows the BLDC under OC fault conditions; (a) current, (b) voltage, (c) speed, and (d) torque performance by varying the time. The current and voltage are not within the exact range at the time interval because of an OC fault. Here, in the BLDC design, the switch S_1 occurs in an OC fault, as shown in Fig. 3. The extracted frequency from the DWT relates to the fault frequency range. The speed and torque of the motor will not be in the constant range because the current flowing from the source to the switch S_1 is open, and thus, an OC fault occurs in the BLDC drive. In the BLDC, when the current flow is not within the normal range, the speed and torque fluctuate.

Table 1 shows the performance under no-fault, SC, and OC fault conditions with respect to speed. Here, the speed rise time is at the minimum level (0.0052) for SC faults, while the settling time is at the minimum level (0.7450) for OC faults. The overshoot time is at the maximum (9.2146×10^{8}) for OC faults, while the undershoot time is at the maximum level (1.6289×10^{3}) for OC faults and at the minimum level (2.3754×10^{-6}) for SC faults. The speed peak time is at the minimum level (0.0254) under no-fault conditions.

Table 2 shows the performance of the no-fault, SC, and OC faults with respect to torque. In this case, the rise time is at the minimum level (2.9957×10^{-8}) for OC faults and the settling time at the minimum level (0.7450) for OC faults. The overshoot time is at the maximum level (1.5133×10^{23}) for SC faults, while the undershoot time is at the maximum level (3.9161×10^{25}) for SC faults. Finally, the peak time is at the minimum level (0.0243) in no-fault conditions.

Table 3 presents a performance comparison of the proposed method with respect to sensitivity, accuracy, and specificity. Here, the performance of the hybrid classifier using the hybrid optimization approach achieves the

***Table 3**: Performance comparison of the proposed method on the basis of accuracy, sensitivity, and specificity.*

Methods	Sensitivity	Specificity	Accuracy
SVM-TLBO	91.6	95.8	93.7
SVM-CPSO	91.1	95.5	92.6
SVM-TLBO+CPSO	97.2	96.3	95.2
NB-TLBO	79.6	89.8	84.7
NB-CPSO	83.1	91.4	87.2
NB-TLBO+CPSO	83.8	91.9	87.8
SVM+NB-TLBO	98.3	98.4	97.9
SVM+NB-CPSO	98.2	97.2	98.4
SVM+NB-TLBO+CPSO	99.5	98.2	98.9

***Table 4**: Benchmark function comparison of the CPSO-TLBO with other optimization algorithms for D = 10.*

Functions	PSO		TLBO		CPSO-TLBO	
	Mean	SD	Mean	SD	Mean	SD
Sphere	9.9×10^{-1}	5.9×10^{-1}	7.0×10^{-13}	1.0×10^{-12}	6.87×10^{-21}	2.1×10^{-20}
Rosenbrock	4.2	6.8×10^{1}	7.4	5.6×10^{-1}	6.8	4.8×10^{-1}
Schwefel P1.2	11.5	4.8	1.0×10^{-12}	2.9×10^{-12}	4.1×10^{-19}	1.7×10^{-18}
Rastrigin	1.4×10^{1}	5.3	1.43×10^{1}	1.5×10^{1}	5.5×10^{-1}	2.6
Ackley	9.2×10^{-2}	2.8×10^{-1}	2.2×10^{-7}	1.1×10^{-7}	1.2×10^{-9}	1.4×10^{-9}

***Table 5**: Benchmark function comparison of the CPSO-TLBO with other optimization algorithms for D = 30.*

Functions	PSO		TLBO		CPSO-TLBO	
	Mean	SD	Mean	SD	Mean	SD
Sphere	4.9×10^{1}	1.3×10^{1}	1.7×10^{-12}	2.2×10^{-12}	3.4×10^{-19}	1.3×10^{-18}
Rosenbrock	2.3×10^{1}	3.5	2.7×10^{1}	7.4×10^{-1}	2.4×10^{1}	5.6×10^{-1}
Schwefel P1.2	3.3×10^{3}	9.5×10^{2}	5.9×10^{-12}	1.0×10^{-11}	5.8×10^{-19}	1.2×10^{-18}
Rastrigin	9.4×10^{1}	2.9×10^{1}	3.1×10^{1}	7.0×10^{1}	8.02×10^{-7}	3.9×10^{-6}
Ackley	3.5	2.7×10^{-1}	3.7×10^{-7}	3.1×10^{-7}	1.2×10^{-9}	9.7×10^{-9}

best accuracy (98.9%), sensitivity (99.5%), and specificity (98.2%). Thus, the proposed methodology offers the best overall performance.

Table 4 shows a benchmark function comparison of the CPSO-TLBO with other optimization algorithms for dimension 10. The mean and standard deviation of PSO, TLBO, and CPSO-TLBO with respect to benchmark functions such as Sphere, Rosenbrock, Schwefel P1.2, Rastrigin, and Ackley are discussed. The proposed method demonstrates the best performance in all functions.

Table 5 shows a benchmark function comparison of the CPSO-TLBO with other optimization algorithms for dimension 30. The mean and standard deviation of PSO, TLBO, and CPSO-TLBO with respect to the benchmark functions such as Sphere, Rosenbrock, Schwefel P1.2, Rastrigin, and Ackley are discussed. The proposed method demonstrates the best performance in all functions.

4.6 Convergence Analysis

The convergence analyses of the proposed method and conventional methods such as TLBO and CPSO are given below.

Fig. 9 represents the convergence analysis of the proposed method and existing methods such as TLBO and CPSO. At the 100^{th} iteration, the fitness of TLBO is in the range of 22, while CPSO has a fitness value in the range of 28, and the proposed CPSO-TLBO a fitness value in the range of 21. In all iterations, CPSO-TLBO demonstrates a higher fitness value than the other methods. Therefore, the proposed CPSO-TLBO is proven to converge faster, while the other two methods take more time.

5. CONCLUSION

This work introduces the diagnosis of OC and SC flaws in BLDC using a hybrid classifier with hybrid optimization, evaluated by MATLAB/SIMULINK under OC, SC fault, and no-fault conditions. Features such

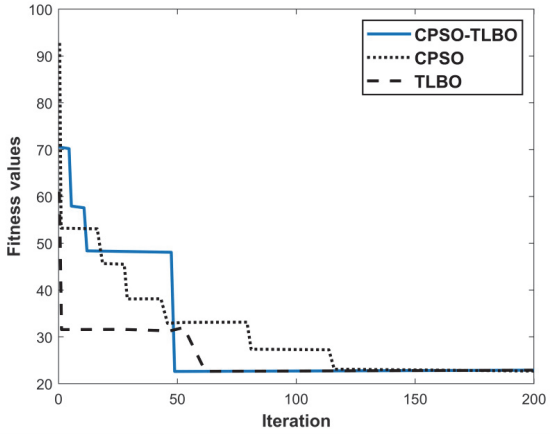

***Fig. 9**: Convergence analysis.*

as current, voltage, speed, and torque are considered as the training data. Here, feature extraction by DWT and classification by SVM-NB were successfully performed. Furthermore, the performance of the system is improved through the employment of the hybrid algorithm TLBO-CPSO. The proposed design is capable of detecting and recognizing the type of faults in the BLDC motor. These hybrid algorithms provide better performance when compared with other approaches with respect to sensitivity, accuracy, and specificity. This improved model achieves an accuracy of about 98.8%. The performance of the CPSO-TLBO is compared with CPSO and TLBO with respect to the benchmark function for dimensions 10 and 30. In the convergence analysis, CPSO-TLBO also proved its superiority. The reliability and robustness of the BLDC can be further enhanced at a lower cost. In future, this methodology will be applied to other types of faults and tested in various conditions. Furthermore, the classification can be carried out by deep learning models.

REFERENCES

[1] S. Lu and X. Wang, "A new methodology to estimate the rotating phase of a BLDC motor with its application in variable-speed bearing fault diagnosis," *IEEE Transactions on Power Electronics*, vol. 33, no. 4, pp. 3399–3410, Apr. 2018.

[2] J. Choi, J.-H. Lee, Y.-G. Jung, and H. Park, "Enhanced efficiency of the brushless direct current motor by introducing air flow for cooling," *Heat and Mass Transfer*, vol. 56, no. 6, pp. 1825–1831, Jun. 2020.

[3] T. A. Shifat and J. W. Hur, "An effective stator fault diagnosis framework of BLDC motor based on vibration and current signals," *IEEE Access*, vol. 8, pp. 106 968–106 981, 2020.

[4] C. He and T. Wu, "Permanent magnet brushless DC motor and mechanical structure design for the electric impact wrench system," *Energies*, vol. 11, no. 6, p. 1360, 2018.

[5] K. Kudelina, B. Asad, T. Vaimann, A. Belahcen, A. Rassõlkin, A. Kallaste, and D. V. Lukichev, "Bearing fault analysis of BLDC motor for electric scooter application," *Designs*, vol. 4, no. 4, p. 42, 2020.

[6] R. L. V. Medeiros, A. C. L. Filho, J. G. G. S. Ramos, T. P. Nascimento, and A. V. Brito, "A novel approach for speed and failure detection in brushless DC motors based on chaos," *IEEE Transactions on Industrial Electronics*, vol. 66, no. 11, pp. 8751–8759, Nov. 2019.

[7] D. A. Papathanasopoulos and E. D. Mitronikas, "Fault tolerant control of a brushless DC motor with defective position sensors," in *2018 XIII International Conference on Electrical Machines (ICEM)*, 2018, pp. 1503–1509.

[8] H. Wang, S. Lu, G. Qian, J. Ding, Y. Liu, and Q. Wang, "A two-step strategy for online fault detection of high-resistance connection in BLDC motor," *IEEE Transactions on Power Electronics*, vol. 35, no. 3, pp. 3043–3053, Mar. 2020.

[9] A. G. Espinosa, J. A. Rosero, J. Cusido, L. Romeral, and J. A. Ortega, "Fault detection by means of hilbert–huang transform of the stator current in a PMSM with demagnetization," *IEEE Transactions on Energy Conversion*, vol. 25, no. 2, pp. 312–318, Jun. 2010.

[10] S. M. R. Balaji, C. Muniraj, and M. N, "Wavelet transform based fault diagnosis of BLDC motor drive," *Indonesian Journal of Electrical Engineering*, vol. 14, no. 3, pp. 434–440, Jun. 2015.

[11] V. M. Fico, M. A. M. Prats, and A. L. R. Vazquez, "Brushless DC motors failure detection using the continuous wavelet transform," *Aerotecnica Missili & Spazio*, vol. 93, no. 3/4, pp. 61–67, Jul. 2014.

[12] A. A. Obed and A. K. Kadhim, "Speed and current limiting control strategies for BLDC motor drive system: A comparative study," *International Journal of Advanced Engineering Research and Science*, vol. 5, no. 2, pp. 119–130, Feb. 2018.

[13] S. R. Vippala, S. Bhat, and A. A. Reddy, "Condition monitoring of BLDC motor using short time Fourier transform," in *2021 IEEE Second International Conference on Control, Measurement and Instrumentation (CMI)*, 2021, pp. 110–115.

[14] C.-Y. Lee and T.-A. Le, "Optimised approach of feature selection based on genetic and binary state transition algorithm in the classification of bearing fault in BLDC motor," *IET Electric Power Applications*, vol. 14, no. 13, pp. 2598–2608, Dec. 2020.

[15] F. Alvarez-Gonzalez, A. Griffo, and B. Wang, "Permanent magnet synchronous machine stator windings fault detection by hilbert–huang transform," *The Journal of Engineering*, vol. 2019, no. 17, pp. 3505–3509, Jun. 2019.

[16] Q. Wu, Z. Ma, G. Xu, S. Li, and D. Chen, "A novel neural network classifier using beetle antennae search algorithm for pattern classification," *IEEE Access*, vol. 7, pp. 64 686–64 696, 2019.

[17] P. Ghamisi and J. A. Benediktsson, "Feature selec-

tion based on hybridization of genetic algorithm and particle swarm optimization," *IEEE Geoscience and Remote Sensing Letters*, vol. 12, no. 2, pp. 309–313, Feb. 2015.

[18] M. Unal, M. Onat, M. Demetgul, and H. Kucuk, "Fault diagnosis of rolling bearings using a genetic algorithm optimized neural network," *Measurement*, vol. 58, pp. 187–196, Dec. 2014.

[19] R. Ilka and S. A. Gholamian, "Application of artificial bee colony algorithm and finite element analysis for optimum design of brushless permanent magnet motor," *IIUM Engineering Journal*, vol. 13, no. 1, Apr. 2012.

[20] S. K. Pandey, C. Bera, and S. S. Dwivedi, "Design of robust PID controller for DC motor using TLBO algorithm," in *2020 IEEE International Conference on Advances and Developments in Electrical and Electronics Engineering (ICADEE)*, 2020.

[21] O. Zandi and J. Poshtan, "Fault diagnosis of brushless DC motors using built-in hall sensors," *IEEE Sensors Journal*, vol. 19, no. 18, pp. 8183–8190, Sep. 2019.

[22] E. Aker, M. L. Othman, V. Veerasamy, I. bin Aris, N. I. A. Wahab, and H. Hizam, "Fault detection and classification of shunt compensated transmission line using discrete wavelet transform and naive bayes classifier," *Energies*, vol. 13, no. 1, p. 243, Jan. 2020.

[23] S. M. Hosseini, F. Hosseini, and M. Abedi, "Stator fault diagnosis of a BLDC motor based on discrete wavelet analysis using ADAMS simulation," *SN Applied Sciences*, vol. 1, no. 11, p. 1406, 2019.

[24] T. A. Shifat and J.-W. Hur, "ANN assisted multi sensor information fusion for BLDC motor fault diagnosis," *IEEE Access*, vol. 9, pp. 9429–9441, 2021.

[25] H. Cherif, A. Menacer, B. Bessam, and R. Kechida, "Stator inter turns fault detection using discrete wavelet transform," in *2015 IEEE 10th International Symposium on Diagnostics for Electrical Machines, Power Electronics and Drives (SDEMPED)*, 2015, pp. 138–142.

[26] W. Feng, J. Sun, L. Zhang, C. Cao, and Q. Yang, "A support vector machine based naive bayes algorithm for spam filtering," in *2016 IEEE 35th International Performance Computing and Communications Conference (IPCCC)*, 2016.

[27] K. Korovkinas, P. Danėnas, and G. Garšva, "SVM and naïve bayes classification ensemble method for sentiment analysis," *Baltic Journal of Modern Computing*, vol. 5, no. 4, pp. 398–409, Dec. 2017.

[28] F. Zou, D. Chen, and J. Wang, "An improved Teaching-Learning-Based Optimization with the social character of PSO for global optimization," *Computational Intelligence and Neuroscience*, vol. 2016, p. 4561507, 2016.

[29] R. Rao, V. Savsani, and D. Vakharia, "Teaching–learning-based optimization: A novel method for constrained mechanical design optimization problems," *Computer-Aided Design*, vol. 43, no. 3, pp. 303–315, Mar. 2011.

K. V. S. H. Gayatri Sarman received his B.Tech. Degree in Electronics & Communication Engineering from Jawaharlal Nehru Technological University, Hyderabad, India, in 2009 and M.Tech. Electronics & Communication Engineering in VLSI & ES from Vasavi Engineering College, Tadepalligudem, India, in 2011. Presently, he is pursuing his Ph.D. Electronics & Communication Engineering in Jawaharlal Nehru Technological University, Kakinada, India, in the area of Embedded systems. Currently, he is working as Assistant professor in Vishnu Institute of Technology, Bhimavaram, India.

Tenneti Madhu received his B.E. degree in Electronics & Communication Engineering from University of Madras, India, in 1992, M.Tech. Electronics & Communication Engineering from Regional Engineering College Kurukshetra, India, in 1994 and Ph.D. Electronics & Communication Engineering from Osmania University, India, in 2004. He has around 25 years of Teaching experience in various organizations. He is a Fellow of IETE and IE (India). He has published around 140 papers in various national and international journals and conferences.

A. Mallikarjuna Prasad received his B.Tech. in Electronics & Communication Engineering from Nagarjuna University, India, in 1988. He received his M.Tech. in Electronics & Instrumentation from Andhra University, India, in 1992 and his Ph.D. in 2009 from Jawaharlal Nehru Technological University, India in the field of Antennas. He is working as a Professor in Electronics & Communication Engineering department at Jawaharlal Nehru Technological University Kakinada, India. He has 25 publications in various international and national journals and conferences. He is a Life Member of Society of EMC Engineers (India), Indian Society for Technical Education (ISTE), Institute of Electronics and Telecommunication Engineers (IETE), and Instrument Society of India (ISI). He won best teacher award by student evaluation of 2008 batch.