Hybrid bacteria foraging-particle swarm optimization algorithm in DTC performance improving for induction motor drive

Salah Eddine Rezgui, Hocine Benalla, Houda Bouhebel
LEC-Laboratory, Department of Electrical Engineering, Frères Mentouri Constantin 1 University, Algeria

Article Info

Article history:

Received Oct 12, 2020
Revised Mar 23, 2021
Accepted Mar 30, 2021

Keywords:

Bacteria foraging
Direct torque control
Hybrid algorithm
Induction motor drive
Particle swarm optimization

ABSTRACT

This paper presents a hybrid algorithm that combines the particle swarm optimization method with the bacteria foraging technique, named: BF-PSO. The aim is to achieve more efficient and precise parameters determination of the regulators that leads to performance improvement in the speed-loop control of an induction motor (IM) implemented in a direct torque control (DTC). The approach consists of tuning the proportional-integral (PI) parameters that meet high dynamics and tracking behavior using the hybrid BF-PSO algorithm. Investigations have been completed with Matlab/Simulink and several performance tests are conducted. The comparison results are exposed with the most used indices in the controllers' tuning with optimization techniques. It will be shown that the presented technique presents better quality results compared to the conventional method of calculated PI.

Corresponding Author:

Salah Eddine Rezgui
Laboratory of Electrical engineering of Constatntine (LEC)
Department of Electrical Engineering
Frères Mentouri Constantin 1 University
Rue Ain Bey Constantin University, Constantine 25000, Algeria
Email: rezgui.salaheddine@lec-umc.org

1. INTRODUCTION

The conventional closed-loop speed regulation in the direct torque (DTC) induction motors control uses calculated PI techniques to achieve an acceptable level of performance. However, although PID controllers are widely used in industrial process control due to their simple structure and easy implementation, they present difficulties and become no longer suitable for the process to be regulated if confronted with external and internal uncertainties including the parametric variation. DTC also has various vulnerabilities, such as large torque ripple, variable switching frequency, and acoustic sound, which are directly or indirectly related to the PI gain values. Thus, the accurateness of these values becomes a crucial deal to the controller designer to conducts a stable system and achieves high performance.

Despite several deep control theories that have been used, such as the Hinf technique as in [1], where a robust controller is designed using hybrid current-flux d-q equations of an induction machine. Additionally, the variable structure control (VSC) known for its robustness, may very well be suitable for systems whose uncertainties are limited, in [2] the authors have used a vector control strategy based on speed sliding mode control (SMC) and torque PI control for construction machinery. But as is familiar in the control community, this technique suffers from the phenomenon of chattering.

The introduction of computational intelligence in the field of process control has allowed numerous research projects to take place, and thus, enabling the development and implementation of new disciplines. The field generally covers bio-inspired techniques that deal with intelligence process, the most known are: fuzzy systems, artificial neural networks (ANN), evolutionary computation, and swarm intelligence (particle swarm (PSO), ant colonie (ACO), and bee colonie optimization (BCO). Since these types do not require a precise model of the process, they have found very wide applications in the field of IM control. Among original works we can found [3], [4] in fuzzy logic, and [5] in the field of control with ANN, or methods that combine both as in [6]. However, all these approaches suffer from computational burden and their use remains closely related to the development of computers and electronics.

In these last two decades, the introduction of meta-heuristic optimization techniques in this field has opened up a new horizon in tuning methods. beneath them, the methods that are based on social and biological behaviors to locate an optimal solution such as; the genetic algorithm which is based on the theory of evolution and natural selection (darwinian/mendelian) resulting from the fittest individuals (survivors) [7], or the cerebral emotional learning mechanism responsible for processing emotions in the brain, this is essentially based on a selection of actions based on sensory inputs and emotional signals, like the regulators implemented in [8] where authors have proposed a brain emotional learning intelligent control for precise speed tracking of a hybrid stepper motor. In another work [9], a differential evolution algorithm was applied to optimize the gains of the PI controller involved in the model predictive torque control that minimizes the speed error. By this strategy, the authors have concluded that the motor's speed response was fast and stable.

There are many other algorithms and classes of algorithms that are inspired by the "individual" and "collective" intelligence of social insects, as well as other animals and fish societies. They use different types of strategies to search for valuable foraging, group re-location, or prey evading. Among many interesting techniques, we can cite ACO and BCO colonies, and bacteria foraging algorithms [10]-[12].

The PSO algorithm was originally introduced by [13], it has acquired its reputation thanks to its simplicity of programming and its adaptation to a large number of problems. Important of recent researches used this technique as in [14], [15]. More specific publications are found in electric machine drives; in [16], a multi-objective particle swarm optimization algorithm was used in vector control drive to improve PI parameters of the speed controller to achieve a fast response of rotor speed and reduce torque ripple. Furthermore, the techniques which use hybridization occupy more and more place in the literature, as in [17] where the paper presents the PSO algorithm in conjunction with the fuzzy logic method to achieve an optimized tuning of a PID controller in a DTC control scheme. Or in [18] where the authors presented a speed and voltage PI controller's tuning algorithm using GA-PSO in vector control of an IM. A summarizing review on this topic can be found in [19].

In this work, we have used a combination of bacteria foraging with particle swarm optimization named: BF-PSO algorithm. The aim is to exploit it for an extensive search to realize a PI parameters optimization to achieve an optimal solution and contribution in the improvement of induction motor drive with DTC. The proposed procedure consists of tuning PI controllers' parameters using Simulink. In this late, we will take into consideration all the DTC and IM dynamics including nonlinératies and the switches model. Multiple manipulations with performance indices are tested; integral of the time multiplied by the absolute value of the error (ITAE), integral of the squared error (ISE), and the integral of the absolute value of the error (IAE), and lastly, the use of the integral time squared error (ITSE) plus the term of overshoot gave better results.

This paper is structured as the following: in Section 2, the model of the induction motor and the DTC theory are presented. In Section 3, the description of PSO and BF techniques is given and followed by the design of the control strategy system based on the BF-PSO technique. The simulation results for the speed tracking and performance tests are given in Section 4, also comparisons of the PI and BF-PSO methods are effectuated with different indices. Finally, conclusions are presented in the Section 5.

2. INDUCTION MOTOR MODEL AND DTC FUNDAMENTALS

Explaining the term direct control of torque and flux is based on the fact that from the errors between the reference values of the torque (and the flux) and those estimated, it is possible if it knows the flux angle (thus the sectors) to directly control the states of the voltage source inverter (VSI) to reduce errors within the hysteresis band controllers as shown on Figure 2.

2.1. Induction motor model

The dynamic model of 3-ph, Y-connected induction motor is given in the d-q synchronous frame as [20]:

$$\begin{cases} \dfrac{di_s}{dt} = -a_1 i_s + \omega_s C i_s + a_2 \omega C \psi_r + a_0 v_s \\ \dfrac{d\psi_r}{dt} = -a_5 i_s + a_4 \psi_r + \omega_{sl} C \psi_r \end{cases} \tag{1}$$

Where $v_s=[v_{sd}, v_{sa}]^T$, $\psi_r=[\psi_{rd}, \psi_{rd}]^T$, and $i_s=[i_{sd}, i_{sq}]^T$ are respectively the vectors of stator voltages, rotor flux linkages, and stator currents. ω_s is the synchronous angular speed, ω is the electrical angular speed of the rotor, and $\omega_{sl} = \omega_s - \omega$ is the slip frequency, and:

$$a_0 = 1/\sigma L_s, a_1 = a_0(R_s + R_r L_m^2 / L_r^2), a_2 = a_0 R_r L_m / L_r^2,$$
$$a_3 = a_0 L_m / L_r, a_4 = R_r / L_r,\ a_5 = a_4 L_m,\ C = \begin{pmatrix} 0 & 1 \\ -1 & 0 \end{pmatrix}.$$

R_s and R_r are the stator and rotor resistances, L_s and L_r are the stator and rotor inductances, L_m is the mutual inductances between the stator and the rotor winding, $\sigma = 1 - (L_m^2/L_r L_s)$ is the total leakage factor. The electromagnetic torque is then expressed as a function of the stator currents and rotor flux components as:

$$T_e = p \frac{3}{2} \frac{L_m}{L_r} (\psi_{rd} i_{sq} - \psi_{rq} i_{sd}) \tag{2}$$

p: is the number of pole pairs.

The mechanical equation of the IM is given by:

$$J\, d\Omega/dt = T_e - T_L - f_v \Omega \tag{3}$$

Where; T_L is the load torque, f_v is the viscous friction coefficient, Ω is the mechanical rotor speed, and J is the inertia moment.

2.2. Direct torque control theory

The DTC development is carried out on the stationary reference fame (α,β), the electrical equations of the IM are then [21]:

$$\begin{cases} v_s = R_s i_s + \dfrac{d\psi_s}{dt} \\ 0 = R_r i_r + \dfrac{d\psi_r}{dt} - j\omega\psi_r \end{cases} \tag{4}$$

Where $\mathbf{i_s}=[i_{s\alpha}, i_{s\beta}]$, $\boldsymbol{\psi_s}=[\psi_{s\alpha}, \psi_{s\beta}]$, $\boldsymbol{\psi_r}=[\psi_{r\alpha}, \psi_{r\beta}]$, $\mathbf{v_s}=[v_{s\alpha}, v_{s\beta}]$ are respectively the vectors and components of; stator currents, stator/rotor flux linkages, and stator voltages.

Stator- flux components are estimated by:

$$\psi_s = \int (v_s - R_s i_s)\, dt \tag{5}$$

Modulus and angle of flux can be obtained as follows:

$$\begin{cases} |\psi_s| = \sqrt{\psi_{s\alpha}^2 + \psi_{s\beta}^2} \\ \theta_s = \tan^{-1}(\psi_{s\beta} / \psi_{s\alpha}) \end{cases} \tag{6}$$

Therefore, the electromagnetic torque is estimated through (7):

$$T_e = p(\psi_{s\alpha} i_{s\beta} - \psi_{s\beta} i_{s\alpha}) \tag{7}$$

Using two hysteresis controllers, the difference between requested and estimated values are evaluated and thereby determine if the flux and torque vectors should be increased, decreased, or constant. Boolean signals are constructed from the switching table and used to genarate six input voltage vectors via the voltage source inverter depending on the sector number, as shown in Figure 1.

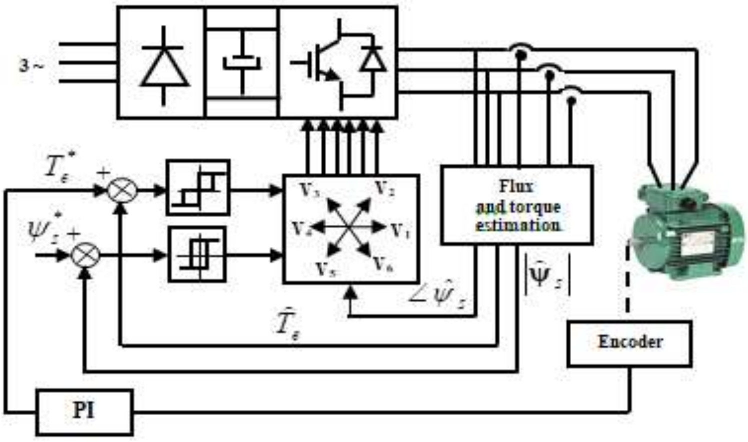

Figure 1. Schematic of the DTC

3. DESCRIPTION OF PARTICLE SWARM OPTIMIZATION AND BACTERIA FORAGING TECHNIQUES

These techniques utilize concepts borrowed from the field of social behavior. They use a cognitive coherence developed to give a solution by acquired personal experience and collective influence of other members of the social group.

3.1. Basic concepts of particle sswarm optimization

At each step (k), each particle moves in a way so it reaches a better local solution (evaluated by a fitness), it remembers the position where it achieved the best value compared to previous searches. This is called the individual best position ($P_{iD}^{(k)}$). Also, the group keeps track of the position where the best value of the whole swarm was reached, and that what it's called the global best position ($P_g^{(k)}$) .

The position and velocity of the i_{th} particle in the k_{th} iteration are denoted by vector of D-dimension $X_{iD}^{(k)}$and $V_{iD}^{(k)}$respectively. The PSO algorithm is performed on the basis of the following two iterative equations [22]:

$$V_{iD}^{(k+1)} = wV_{iD}^{(k)} + c_1r_1(P_{iD}^{(k)} - X_{iD}^{(k)}) + c_2r_2(P_{gD}^{(k)} - X_{iD}^{(k)}) \tag{8}$$

$$X_{iD}^{(k+1)} = X_{iD}^{(k)} + V_{iD}^{(k+1)} \tag{9}$$

Where; w denotes the inertia weight, it's used to provide a balance between global and local search, thus requiring less iteration on average to find a sufficiently optimal solution, it's set according to (10), and decreases linearly from about 0.9 to 0.4 [23]. c_1 is the cognitive and c_2 is the social parameter. The coefficients r1, r2 are random numbers belonging to 0 and 1.

$$w = w_{max} - Iter.(W_{max} - W_{min}) / Iter_{max} \tag{10}$$

Where; Iter, and $Iter_{max}$, are the current iteration, and the maximum number of iterations respctively.

On the right-hand side of (8), the second segment refers to the cognitive part, and it represents the distance between the particle $X_{iD}^{(k)}$ and its best-located solution $P_{iD}^{(k)}$. While the third segment represents the social component, which reflects the distance between the same particle and the global best solution $P_{gD}^{(k)}$. Once the velocity vector is updated based on the individual history and collective experience of the swarm, then, the particle moves to a new position through (9). This procedure continues until the best solution is reached or the algorithm meets user-defined stopping criteria.

3.2. Bacteria foraging description

Bacteria Foraging technique is developed from a nature-inspired optimization algorithm, it's mainly based on the foraging behavior as a group of Escherichia coli bacteria (E. coli).

3.2.1. Escherichia coli (E. coli) bacteria [24]

The E. coli bacteria have a control system that allows them to search for food (higher graduation of the nutrient) and to avoid harmful substances. A set of flagella gives bacteria the ability to move around by "swimming" or "tumbling", and generate consequently a motion pattern (called "chemotaxis") based on the presence of chemical attractants and repellents.

Elimination and dispersal are part of motile behavior at the population level. The local environment in which a population of bacteria lives may change, in consequence, there will be events such as all the bacteria in a region are killed or a group is dispersed in a new part of the environment. It has the effect of damaging the chemotactic development, however, it also has a possible counter-effect (stimulating the chemotaxis) since the dispersion can relocate bacteria towards good sources of food.

It is noted that E. coli bacteria besides chemotaxis is capable of "thermotaxis" in that it seeks warmer environments and "phototaxis" since it tries to avoid intense blue light, and can develop some kind of reproduction called "conjugation," and a mutation rate. A particularly group behavior has been demonstrated at high levels of the nutrient, the bacteria release an attractant so that they congregate into groups and hence, move as concentric patterns of groups with high bacterial density.

3.2.2. Mathematical construction of bacterial foraging algorithm

To model the BF algorithm we need to define a population (S) of bacteria that execute these main actions; chemotaxis, swarming and tumbling, reproduction, and elimination-dispersion.First, let define J(i,j,k,l) the function coast value, $\theta^i(j,k,l)$ the i^{th} bacteria position, and P(j,k,l) which represents the positions of each member in the population S at the j^{th} chemotactic step, k^{th} the reproduction step, and $l^{th}$the elimination-dispersal event.Other BF parameters are defined in Table 1 [24].

Table 1. The parameters of bacteria foraging algorithm

Variable	Definition
Nc	Number of chemotactic steps
Nre	The number of reproduction steps
Ns	Maximum number of steps (swim length)
Ned	Elimination-dispersal events
Ped	Elimination-dispersal probability
C(i)	The step size during runs, i=1,2,..........,S
φ(i)	Represents a tumble (in random direction)
n	Dimension of the search space
Δ(i)	Random vector

a) Population and chemotaxis:

Themovement of i^{th} bacterium after one step can be expressed by [24]:

$$\theta^i(j+1,k,l) = \theta^i(j,k,l) + C(i)\frac{\Delta(i)}{\sqrt{\Delta^T(i)\Delta(i)}} \tag{11}$$

If at $\theta^i(j+1,k,l)$ the cost J(i, j +1,k,l) is better (lower) than at $\theta^i(j, k, l)$, then another step of size C(i) in the same direction will be taken. This swim is continued as long as it continues to reduce the cost (up to Ns). In other words, the cell will tend to continue to spread if it moves towards gradually favorable environments.

b) Swarming mechanism:

The bacteria release an attractant so that they should swarm together. The bacterium also repels nearby ones in the sense it will be not other one at the same location. The combined cell-to-cell attraction and repelling effectscan be modeled as [25]:

$$\begin{aligned} J_{cc}(\theta, P(j,k,l)) &= \sum_{i=1}^{S} J_{cc}^i(\theta, \theta^i(j,k,l)) \\ &= \sum_{i=1}^{S}\left[-d_{att}\exp\left(-\omega_{att}\sum_{m=1}^{P}\left(\theta_m-\theta_m^i\right)^2\right)\right] + \sum_{i=1}^{S}\left[-h_{rep}\exp\left(-\omega_{rep}\sum_{m=1}^{P}\left(\theta_m-\theta_m^i\right)^2\right)\right] \end{aligned} \tag{12}$$

Here P is the number of parameters to be optimized and $\theta=[\theta_1....\theta_P]^T$ is a point on the optimization domain and θ^i_m s the m[th] component of the i[th] bacterium position θ^i. d_{att} is the depth of the attractant released by a cell, and ω_{att} is a measure of the width of the attractant signal. h_{rep} and ω_{rep} represent the same quantities but for the repels effects.

c) Reproduction:

When chemotactic steps N_c are attained, a reproduction step is taken with till N_{re}. The half of bacteria having poor fitness (are not "healthy" and thus unlikely to reproduce) die and the second remaining bacteria Sr= S/2 are allowed to reproduce (split in two).

d) Elimination and dispersal:

The bacteria population is subjected to elimination-dispersal with probability factor (P_{ed}). These events (with N_{ed} times) take place after several generations of reproduction. This process may disturb the algorithm but it prevents the population to be stuck on local minima.

3.2.3.Hybrid bacteria foraging with particla swarm optimization procedure (BF-PSO)

Our technique is based on the initial bacteria foraging algorithm (BF) which is enhanced by the PSO technique. This approach was proposed firstly by [26] and subsequently taken up by [27]. The initial algorithm relies on the random generation of the bacteria tumbling direction vector ($\Delta_m(i)$), this maneuver may lead in a delay of the algorithm to converge toward the global solution.

The PSO technique is exploited by its peculiarity of using individual and social information, so the best-local position and the best-global position of each bacterium will influence the random direction of the tumbling activities of the bacteria.

Thus, during the process of chemotaxis loop, the vector of the tumble direction is updated using:

$$V = w.V + C_1.r_1(\theta_{lbest} - \theta_{current}) + C_2.r_2(\theta_{gbest} - \theta_{current})$$
$$\Delta_m(i) = V \tag{13}$$

As is shown in Figure 2, a PSO-optimized new direction $\Delta_m(i)$ is incorporated in the BF technique (Implicit subscribes are intentionally dropped).

4. RESULTS AND DISCUSSION

The performances of the presented method are tested within Matlab/Simulink. Real parameters are used in the simulations, they were obtained by identification procedure in the laboratory of three-phase Y connected squirrel cage induction motor, 1 kW, 2880 rpm, 220/380V, 4/2.3 A, 50Hz [28].

The performance index (or fitness) used in this study is the Integral of the Time multiplied by the Squared Error (ITSE) with an addition of the system overshoot according to the following equation:

$$\begin{aligned} J &= \alpha.ITSE + \beta.overshoot \\ &= \alpha\int te^2(t)\,dt + \beta.overshoot \end{aligned} \tag{14}$$

Where e(t) is the error between the reference and the desired speed:

$$e(t) = \Omega^*(t) - \Omega(t) = N^*(t) - N(t) \tag{15}$$

The calculated gains' values with the conventional PI method are; $K_{pw} = 1.5$, $K_{iw} = 0.1$. The optimal values obtained by the BF-PSO algorithm are: $K_{pw} = 12.6822$, $K_{iw} = 0.1473$.

Figure 3 presents speed tracking curves between reference (N*) and actual rotor speed (N) for both PI and BF-PSO methods. Initially, a step command with 2800 rpm is applied without load, and at t=0.6 s the machine is fully loaded with TL= 3.2 N.m. Then, a negative step reference (-2800 rpm) is applied at t=1s. The zoom-in shows clearly the time response and the disturbance rejection. As we can observe, the speeds reach their references at the same time, but in the case of classic PI regulation, the response shows an important overshoot (5.3%) compared to BF-PSO (0.71%), besides the error in the steady-state.

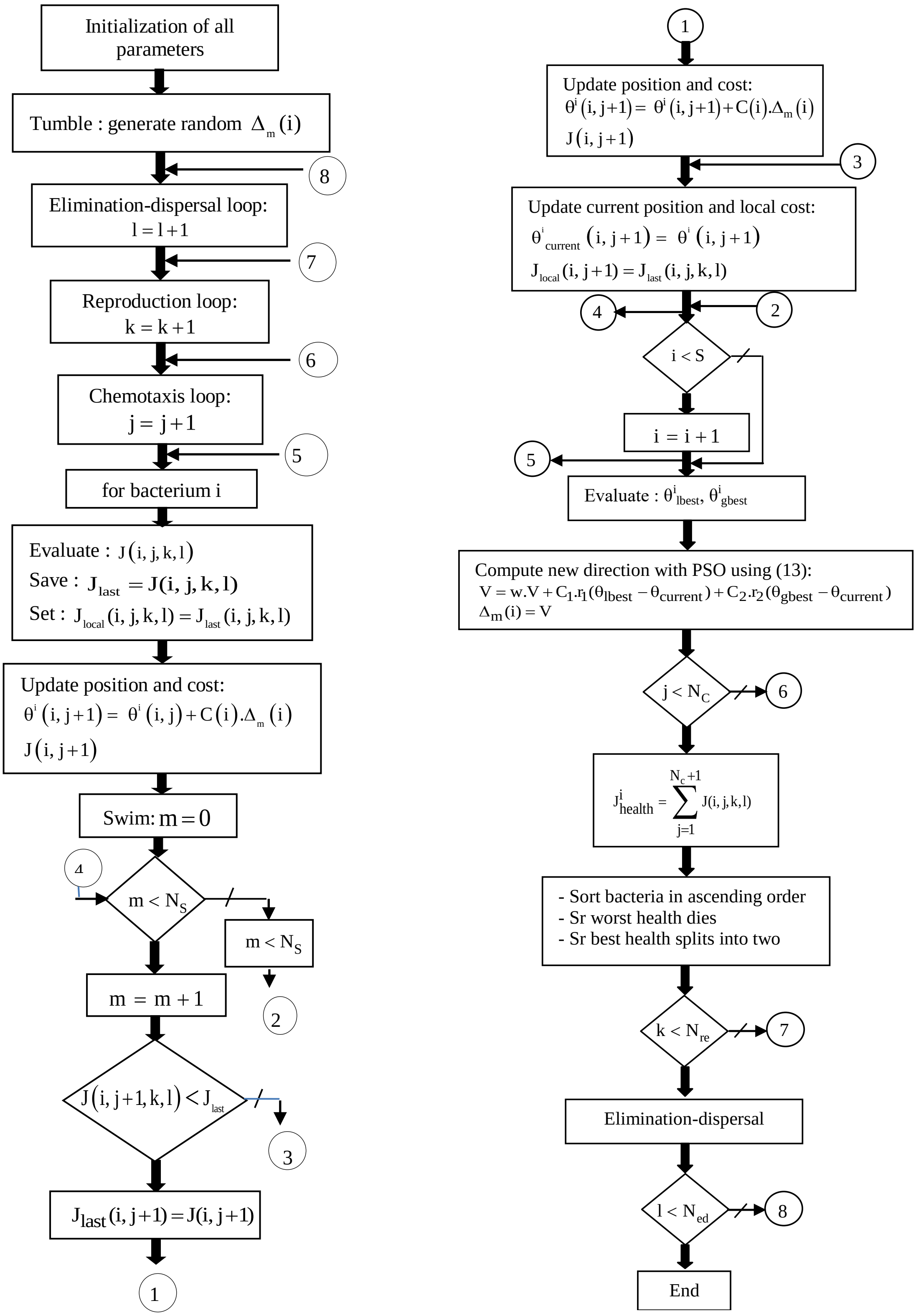

Figure 2. Hybrid bacteria foraging particle swarm optimization algorithm flowchart

We also note that the speed drops due to the application of the load torque and presents a consistent static error in steady-state, whereas with the BF-PSO, the system remains stable. Consequently, the rejection of the disturbance has been improved significantly. In Figure 4, one can observe that the torque presents slightly more ripples when compared with the PI method, but both show the same dynamic.

The flux trajectories exhibit the same curves as shown in Figure 5.Finally, in Figure 6 are shown comparison results obtained with different indices wich are mostly used in controllers tuning via optimization as; ITAE, ISE, IAEgiven in (16):

$$\begin{cases} ISE = \int e^2 dt \\ IAE = \int |e|\, dt \\ ITAE = \int t\, |e|\, dt \end{cases} \tag{16}$$

The curves of speed indicate that the response times are similar, however, the behavior concerning disturbances is clearly better with the adopted index, both when the regulated speed drops and when the time in which the speed returns to its reference.

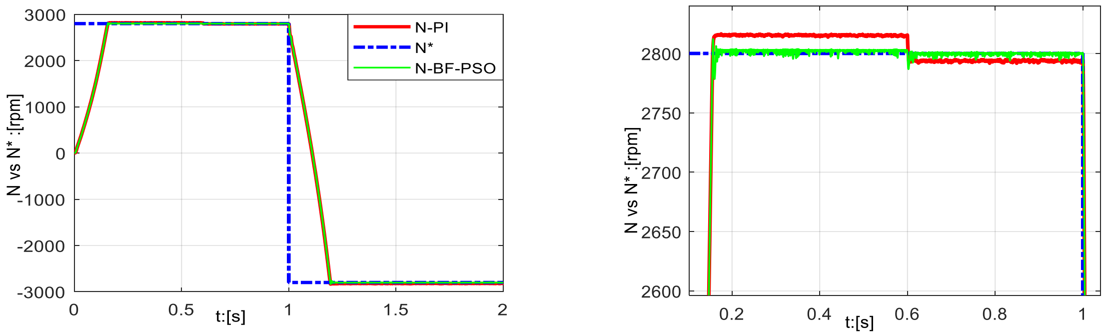

Figure 3. Speed tracking test with conventional PI and BF-PSO algorithm

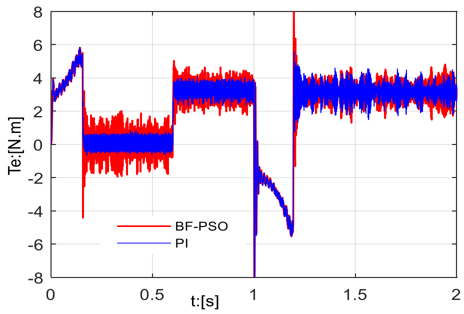

Figure 4. Torque response with PI and BF-PSO algorithm

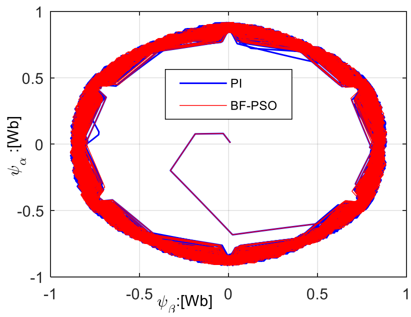

Figure 5. Flux trajectories

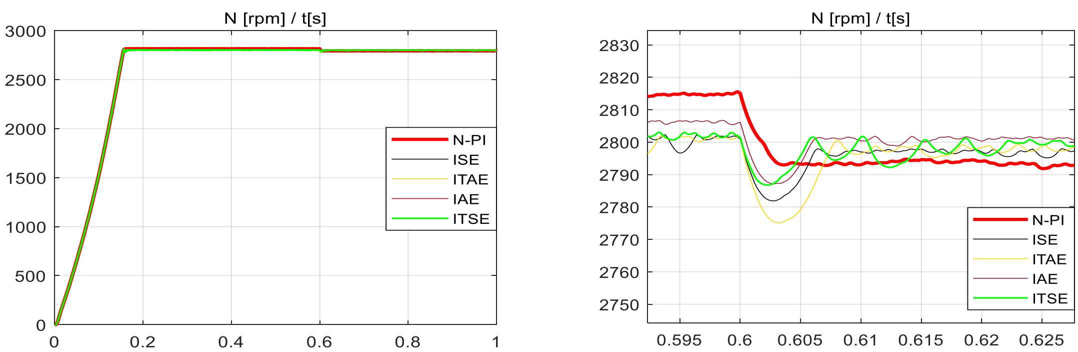

Figure 6. Speed response with different indices used with BF-PSO algorithm

5. CONCLUSION

The performance of DTC with classic PI was tested and compared with the BF-PSO tuned PI controllers. The results have shown that the presented method improves the system stability and robustness against disturbance. The simulation results have shown that with BF-PSO tuned PI, the system's presents good dynamic and more effectiveness in disturbance rejection, with remarkably less overshoot and steady-state error. Results reveal the efficiency of the BF-PSO algorithm to achieve optimal solutions and contribution in the improvement of induction motor drive with DTC in different operating phases (starting, transient, steady-state evaluation).

REFERENCES

[1] Lukas Pohl, Ivo Vesely, "Speed Control of Induction Motor Using H∞ Linear Parameter Varying Controller,"*IFAC-PapersOnLine*, vol. 49, no. 25, pp. 74-79, 2016, doi: 10.1016/j.ifacol.2016.12.013.

[2] S. Fu, H. Ren, T. Lin, S. Zhou, Q. Chen and Z. Li, "SM-PI Control Strategy of Electric Motor-Pump for Pure Electric Construction Machinery,"*IEEE Access,* vol. 8, pp. 100241-100250, 2020, doi: 10.1109/ACCESS.2020.2997517.

[3] N. Jabbour, C. Mademlis, "Online Parameters Estimation and Autotuning of a Discrete-Time Model Predictive Speed Controller for Induction Motor Drives,"*IEEE Transactions on Power Electronics*, vol. 34, no. 2, pp. 1548-1559, 2019, doi: 10.1109/TPEL.2018.2831459.

[4] C. Wang and Z. Q. Zhu, "Fuzzy Logic Speed Control of Permanent Magnet Synchronous Machine and Feedback Voltage Ripple Reduction in Flux-Weakening Operation Region," *IEEE Transactions on Industry Applications*, vol. 56, no. 2, pp. 1505-1517, 2020, doi: 10.1109/TIA.2020.2967673.

[5] H. Jie, G. Zheng, J. Zou, X. Xin, L. Guo, "Speed Regulation Based on Adaptive Control and RBFNN for PMSM Considering Parametric Uncertainty and Load Fluctuation,"*IEEE Access*, vol. 8, pp. 190147-190159, 2020, doi: 10.1109/ACCESS.2020.3031969.

[6] H. Sathishkumar, S.S. Parthasarathy, "A novel neuro-fuzzy controller for vector controlled induction motor drive," *Energy Procedia*, vol. 138, pp. 698-703, 2017, doi: 10.1016/j.egypro.2017.10.203.

[7] S. Haffner, L.A. Pereira, L.F.A. Pereira, "A Method for Optimization of Five-Phase Induction Machines Based on Genetic Algorithms,"*J. Control Autom Electr Syst*, vol. 26, pp. 521–534, 2015, doi: 10.1007/s40313-015-0197-z.

[8] A. M. Yazdani, A. Mahmoudi, M. A. Movahed, P. Ghanooni, S. Mahmoudzadeh, S. Buyamin, "Intelligent Speed Control of Hybrid Stepper Motor Considering Model Uncertainty Using Brain Emotional Learning,"*Canadian Journal of Electrical and Computer Engineering*, vol. 41, no. 2, pp. 95-104, 2018.

[9] A. N. Rosa Lima, G. Adriano Juliani Storti, M. F. Castoldi, A. Goedtel, E. R. Pires da Silva, "Optimization of the PI controller used in Model Predictive Torque Control through Differential Evolution," *IEEE 28th International Symposium on Industrial Electronics (ISIE), Vancouver, BC, Canada*, pp. 348-353, 2019, doi: 10.1109/ISIE.2019.8781449.

[10] Z. Yin, C. Du, J. Liu, X. Sun and Y. Zhong, "Research on Autodisturbance-Rejection Control of Induction Motors Based on an Ant Colony Optimization Algorithm,"*IEEE Transactions on Industrial Electronics*, vol. 65, no. 4, pp. 3077-3094, 2018, doi: 10.1109/TIE.2017.2751008.

[11] F. B. Sharma, S. R. Kapoor, "Induction motor parameter estimation using disrupted black hole bee colony algorithm," *International Journal of Metaheuristics*, vol. 6, no. 1/2, pp.85-106, 2017, doi: 10.1504/IJMHEUR.2017.083098.

[12] V. Souza Santos, P. V. Felipe, J. G. Sarduy, "Bacterial foraging algorithm application for induction motor field efficiency estimation under unbalanced voltages,"*Measurement*, vol. 46, no. 7, pp. 2232-2237, 2013, doi: 10.1016/j.measurement.2013.03.019.

[13] J. Kennedy and R. C. Eberhart, "Particle Swarm Optimization," in *Proc. of the IEEE Int. Conf. on Neural Networks. Piscataway, NJ: IEEE Service Center*, pp. 1942-1948, 1995, doi: 10.1007/978-3-030-59392-6_2.

[14] C. Chakraborty, "Chronic Wound Image Analysis by Particle Swarm Optimization Technique for Tele-Wound Network," Springer, *Int. JournalofWireless Personal Communications,* pp. 3655-3671, 2017, doi: 10.1007/s11277-017-4281-5.

[15] W. R. Abdul-Adheem, "An enhanced particle swarm optimization algorithm,"*International Journal of Electrical and Computer Engineering(IJECE)***,** vol. 9, no.6, pp. 4904-4907, 2019, doi: 10.11591/ijece.v9i6.pp4904-4907.

[16] G. Shashibhushan, Savita Sonoli, "Multi-objective solution with PSO algorithm for minimization of torque ripple and speed settling time by using solar-fed 11, 9, and 3-level multi-level inverter with vector control of induction motor,"*International Journal of Power Electronics and Drive System (IJPEDS),* vol. 11, no. 2, pp. 827-832, 2020, doi: 10.11591/ijpeds.v11.i2.pp827-832.

[17] G. Boukhalfa, S. Belkacem, A. Chikhi, S. Benaggoune, "Direct torque control of dual star induction motor using a fuzzy-PSO hybrid approach," *Applied Computing and Informatics*, 2018, doi: 10.1016/j.aci.2018.09.001.

[18] Dong Hwa Kim, "Tuning Innovation with Biotechnology,"*Pan Stanford Publishing Pte Ltd,* ISBN (eBook): 978-1-315-36458-2, chap.4, 2017, doi: 10.1201/9781315364582.

[19] A. Alwadie, "A Concise Review of Control Techniques for Reliable and Efficient Control of Induction Motor,"*International Journal of Power Electronics and Drive System (IJPEDS),* vol. 9, no.3, pp. 1124-1139, 2018, doi: 10.11591/ijpeds.v9n3.pp1124-1139.

[20] Najib El Ouanjli, S. Motahhir, A. Derouich, A. El Ghzizal, A. Chebabhi, M. Taoussi, “Improved DTC strategy of doubly fed induction motor using fuzzy logic controller,”*Energy Reports*, vol.5, pp.271-279, 2019, doi: 10.1016/j.egyr.2019.02.001.
[21] N.R.N. Idris and A.H.M. Yatim, “Reduced Torque Ripple and Constant Torque Switching Frequency Strategy Direct Torque Control of Induction Machine,”*Applied Power Electronics Conference and Exposition, IEEE,* 2000, doi: 10.1109/APEC.2000.826099.
[22] Y. Shi, R.C. Eberhart, “A modified particle swarm optimizer,” *Proceedings of the IEEE International Conference on Evolutionary Computation*, pp. 69-73, 1998, doi: 10.1109/ICEC.1998.699146.
[23] M. G. Bijan, P. Pillay, “Efficiency Estimation of the Induction Machine by Particle Swarm Optimization Using Rapid Test Data With Range Constraints,”*IEEE Transactions on Industrial Electronics*, vol. 66, no. 8, pp. 5883-5894, 2019, doi: 10.1109/TIE.2018.2873121.
[24] K. M. Passimo, “Biomimicry for Optimization, Control, and Automation,” *Springer-Verlag London Berlin Heidelberg*, ISBN 1-85233-804-0, 2005, doi: 10.1007/b138169.
[25] S. Das, A. Biswas, S. Dasgupta, A. Abraham, “Bacterial Foraging Optimization Algorithm: Theoretical Foundations, Analysis, and Applications,”*In: Abraham A., Hassanien AE., Siarry P., Engelbrecht A. (eds) Foundations of Computational Intelligence Vol. 3. Studies in Computational Intelligence, Springer, Berlin, Heidelberg*, vol. 203, pp. 23-55, 2009, doi: 10.1007/978-3-642-01085-9_2.
[26] W. M. Korani, H. T. Dorrah, H. M. Emara, “Bacterial foraging oriented by Particle Swarm Optimization strategy for PID tuning,” *IEEE International Symposium on Computational Intelligence in Robotics and Automation - (CIRA),* pp. 445-450, 2009, doi: 10.1109/CIRA.2009.5423165.
[27] R. Pradhan, M.R. Kaba, S.P. Sahoo, “A Bacteria Foraging-Particle Swarm Optimization Algorithm for QoS Multicast Routing,”*In: Panigrahi B.K., Suganthan P.N., Das S., Dash S.S. (eds) Swarm, Evolutionary, and Memetic Computing. SEMCCO*, pp. 590-600, 2013, doi: 10.1007/978-3-319-03753-0_53.
[28] S.E. Rezgui, A. Mehdi, S. Legrioui, H. Meddouce, A.M. Boulahia, H. Benalla,“IRFOC vs DTC Performance Comparison Analysis,” *3rd International Conference on Electric Power and Energy Conversion Systems, Istanbul*, pp. 1-6, 2013, doi: 10.1109/EPECS.2013.6713066.