FACTA UNIVERSITATIS
Series: **Electronics and Energetics** Vol. 34, N° 2, June 2021, pp. 203-217
https://doi.org/10.2298/FUEE2102203G

Original scientific paper

PSO BASED TAKAGI-SUGENO FUZZY PID CONTROLLER DESIGN FOR SPEED CONTROL OF PERMANENT MAGNET SYNCHRONOUS MOTOR

Hamid Ghadiri[1], Hamed Khodadadi[2], Hooman Eijei[3], Milad Ahmadi[4]

[1]Faculty of Electrical, Biomedical and Mechatronics Engineering, Qazvin Branch, Islamic Azad University, Qazvin, Iran
[2]Department of Electrical Engineering, Khomeinishahr Branch, Islamic Azad University, Isfahan, Iran
[3]Engineering and Technical Department, Electricity Distribution Company of Lahijan, Lahijan, Iran
[4]School of Control Science and Engineering, Shandong University, Jinan 250061, PR China

Abstract. *A permanent magnet synchronous motor (PMSM) is one kind of popular motor. They are utilized in industrial applications because their abilities included operation at a constant speed, no need for an excitation current, no rotor losses, and small size. In the following paper, a fuzzy evolutionary algorithm is combined with a proportional-integral-derivative (PID) controller to control the speed of a PMSM. In this structure, to overcome the PMSM challenges, including nonlinear nature, cross-coupling, air gap flux, and cogging torque in operation, a Takagi-Sugeno fuzzy logic-PID (TSFL-PID) controller is designed. Additionally, the particle swarm optimization (PSO) algorithm is developed to optimize the membership functions' parameters and rule bases of the fuzzy logic PID controller. For evaluating the proposed controller's performance, the genetic algorithm (GA), as another evolutionary algorithm, is incorporated into the fuzzy PID controller. The results of the speed control of PMSM are compared. The obtained results demonstrate that although both controllers have excellent performance; however, the PSO based TSFL-PID controller indicates more superiority.*

Key words: *Particle Swarm Optimization (PSO), Takagi-Sugeno Fuzzy Logic (TSFL), PID, PMSM, Genetic Algorithm (GA).*

Received September 7, 2020; received in revised form December 2, 2020
Corresponding author: Hamid Ghadiri
Faculty of Electrical, Biomedical and Mechatronics Engineering, Qazvin Branch, Islamic Azad University, Qazvin, Iran
E-mail: h.ghadiri@qiau.ac.ir

1. INTRODUCTION

The PMSM has been broadly used in many industrial applications and rapid replacement of induction and DC motors in servo applications. PMSMs are very favorite due to their efficiency, high power density, underweight, and small size comparing to DC and induction machines [1-3].

Synchronous motors are constant speed machines that the frequency of the armature current determines their speed. The armature current depends on armature voltage. Therefore, the simplest way to control the synchronous motor speed is to use the frequency of the voltage applied to the armature. However, in steady-state conditions, the synchronous motor speed is determined by the excitation frequency, but the frequency control speed is limited practically.

The main reason is that it is difficult for a synchronous machine's rotor to follow an arbitrary change in the voltage frequency applied to the armature. Control of synchronous motors can be facilitated by control algorithms where the stator flux and its relation to the rotor flux are directly controlled. This structure leads to torque control of the synchronous motors. Some control methods such as model predictive control (MPC) [4], terminal sliding mode control [5,6], application of support vector machine in internal model control [7], adaptive control [8], input-output feedback linearization technique [9], and neural network control approach [10,11] are employed to control the speed and position of PMSMs.

However, traditional PID control approaches are well-known yet, whereas these controllers are applicable readily while working well around the desired point [12,13]. Notably, conventional PID controllers fail to guide the system to a high-performance operation due to integral windup.

Also, the model-based control approaches are presented in [14-16]. In [17,18], researchers have explained demagnetization fault diagnosis and fault model recognition as a significant issue on the PMSM derive system. In [19], the presented method aims to reject voltage disturbances with an internal model control (IMC); however, the IMC method fails when the system's parameters vary during their operation significantly. A fuzzy control system is analyzed in [20] with a lack of indicating transient speed responses under the load torque variations. Predictive model methods are considered by several researchers so far. A beneficial MPC method is applied to PSMS derives [21]. As a significant disadvantage in this method, solving the optimization problem requires a burdensome computational process for each instant sampling.

As follow, a PID controller is considered to minimize the error signal; however, the constant PID coefficients are inefficient due to the system's nonlinear behavior. Several studies have been assigned to perform the self-tuning PID coefficients. Fuzzy logic can be identified as the most effective method to offer excellent flexibility for PID coefficients, changes under the system's nonlinear behavior [22]. The combination of fuzzy and PID controllers can create a reliable controller for nonlinear models. However, there are considerable deficiencies in some applications. When the system behavior becomes highly nonlinear, it is virtually impossible to use the PID controller under the fuzzy rules.

This study's presented approach is based on optimizing the fuzzy rules in the Takagi-Sugeno fuzzy logic PID (TSFL-PID) controller to be more flexible than the conventional TSFL-PID controller. The TSFL-PID control parameters are tuned based on PSO. Besides, the TSFL-PID controller is optimized with GA for the comparison. The obtained results indicate that the PSO algorithm performs better than GA.

The rest of this paper is organized as follows. The PMSM model and design procedure of the fuzzy PID controller will be explained in sections 2 and 3. Afterward, the TSFL-PID controller is presented and optimized using PSO and GA in section 4. Section 5 is dedicated

to present the simulation results and the resultant discussion. Finally, this paper is concluded in section 6.

2. MATHEMATICAL MODEL OF PMSM

The mathematical model of PMSM in the $d-q$ coordinates is shown below. The voltage equation is [23]:

$$u_q = R_s i_q + L_q \dot{i}_q + w_e L_d i_d + w_e \Psi_f \,, \tag{1}$$
$$u_d = R_s i_d + L_d \dot{i}_d - w_e L_q i_q \,,$$

where u_d and u_q present the direct and quadrature-axis input voltages, i_d and i_q represent the direct and quadratic-axis currents. Besides, the resistance of the stator phase and inductances of the direct and the quadrature-axis are demonstrated as R_s, L_d as well as L_q, respectively. Furthermore, Ψ_f, and w_e denote the excitation magnetic field and rotor angular speed. The magnetic bond equation is considered by the following equation [24]:

$$\Psi_d = L_d i_d + \Psi_f \,, \tag{2}$$
$$\Psi_q = L_q i_q \,,$$

where Ψ_d, and Ψ_q are the existed magnetic fields in the stator winding of the direct-axis and quadrature-axis, respectively. The d-q coordinates, PMSM electromagnetic torque is [25]:

$$T_e = n_p\big(\Psi_f i_q - \big(L_d - L_q\big) i_q i_d\big) \,, \tag{3}$$

where n_p indicates the pole pairs value. The PMSM motion equations denote as follows:

$$Jp\dot{\Omega}_r = T_e - T_l - B\Omega_r \,, \tag{4}$$
$$\Omega_r = \frac{w_e}{n_p}.$$

where Ω_r is the mechanical angular speed of the rotor, J is the moment of inertia of the rotor and T_l is the load (external) torque. Therefore, the state-space equations can be concluded using the above equations as:

$$\begin{aligned} \dot{i}_q &= \frac{1}{L_q}\big(u_q - R_s i_q + L_d i_d \omega_e - \Psi_f \omega_e\big) \,, \\ \dot{i}_d &= \tfrac{1}{L_d}\big(u_d - R_s i_d + \omega_e L_q i_q\big) \,, \\ \dot{\omega}_e &= \tfrac{1}{J}(1.5 n_p^2\big(\Psi_f i_q + \big(L_d - L_q\big) i_d i_q\big) - n_p T_l - B\omega_e). \end{aligned} \tag{5}$$

In the PMSM vector control system, i_d is assumed to be zero. Thus, the equation of state space (5) can be rewritten as follows:

$$\begin{aligned} \dot{i}_q &= \frac{1}{L_q}\big(u_q - R_s i_q - \Psi_f \omega_e\big) \,, \\ \dot{\omega}_e &= \tfrac{1}{J}(1.5 n_p^2 \Psi_f i_q - n_p T_l - B\omega_e). \end{aligned} \tag{6}$$

A field-oriented vector-controlled isotropic PMSM dynamic equation can be obtained as follow:

$$\dot{\omega}(t) = \varphi_1 i_{qs}(t) - \varphi_2 \omega(t) - \varphi_3 T_L(t), \tag{7}$$

where $\omega = \dot{\theta}$ is the electrical rotor angular speed, θ is the electrical rotor angle, T_L illustrates the load torque disturbance input, and $\varphi_i > 0, i = 1, \ldots, 3$ are as follows:

$$\varphi_1 = \frac{3P^2}{8J}\lambda_m \, , \varphi_2 = \frac{B}{J} \, , \varphi_3 = \frac{P}{2J} \tag{8}$$

while p is the number of poles, J, B, and λ_m are the rotor inertia, the viscous friction coefficient, and the magnetic flux, respectively. Fig. 1 demonstrates the block diagram of the control system applied for PMSM.

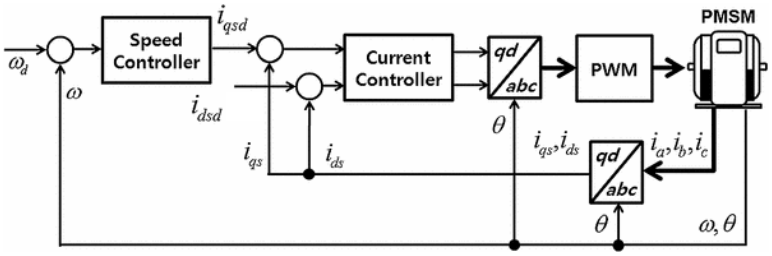


Fig. 1 Block diagram of a field-oriented PMSM control system [26].

As can comprehend, the three-phase current commands can be calculated via converting the controller commands i_{qsd}, i_{dsd}. In which, i_{dsd} assumed as the direct-axis reference current is typically equal to zero. In a field-oriented PMSM control system, the three-phase current commands are computed by converting the current controller commands i_{qsd}, i_{dsd}. Thus, the main challenge can be described as proposing an evolutionary algorithm (EA) based FL-PID for speed control of PMSM.

3. FUZZY-PID CONTROLLER DESIGN

3.1. PID control structure

One of the most widely used controllers in various industries is the PID controller. The main reason for this type's extensive use is its simplicity, where the control signal can be calculated as follows [27]:

$$u(t) = k_p e(t) + k_i \int e(t)d(t) + k_d \frac{d}{dt} e(t) \tag{8a}$$
$$e(t) = r(t) - y(t).$$

where $u(t)$ is the control signal, $e(t)$ is error signal, $r(t)$, and $y(t)$ are reference and output signals, respectively. The proportional, integral, and derivative terms are identified as k_p, k_i, and k_d, respectively. The closed-loop system's operation can be affected directly by the mutation of controller parameters [28].

3.2. Fuzzy logic controller

Uncertainty is one of the inseparable parts of the industrial system [29-31]. The fuzzy logic approach controls a system intelligently without dependency on an accurate system model

[32,33]. A fuzzy inference system (FIS) consists of a fuzzifier, rule base, inference engine, and defuzzifier. A set of fuzzy if-then rules the central part called the knowledge base.

3.3. Designing fuzzy-PID controller

The traditional PID controller should be modified by using a Takagi-Sugeno fuzzy controller to tune the PID gains as follows [26]:

Rule i :

IF ω_e and $\dot{\omega}_e$ is F_i, THEN $i_{qsd} = -K_i^P \omega_e - K_i^I \int_0^t \omega_e d\tau - K_i^D \dot{\omega}_e$ (9)

where $\omega_e = \omega - v_d$, ω_d is the desired speed, $F_i (i = 1, \dots, 2n-1)$ are fuzzy sets, $n > 1$, $r = 2n-1$ is the number of fuzzy rules, and k_i^P, k_i^I, k_i^D are the constants. The fuzzy sets F_i can be determined by the membership function m_i. We assume that the fuzzy set F_n computes for $\omega_e = 0$, F_i calculates for more negative ω_e than F_{i+1} does for $1 \le i \le n-1$, *and* F_{i+1} accounts for more positive ω_e than F_i does for $n \le i \le 2n-2$. Fig. 2 illustrates the membership function of the input variable "ω_e" and $\dot{\omega}_e$ that the inputs are normalized into [-1,1]. In this figure, ZO(F_3) denotes zero, NB(F_1) represents negative big, NM(F_2) is the negative medium, PM(F_4) is a positive medium, and PB(F_5) is positive big.

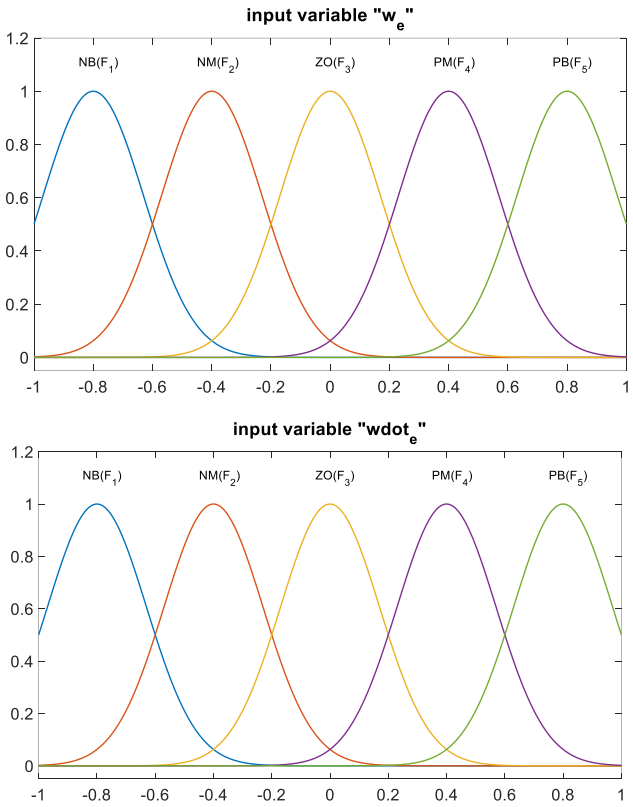


Fig. 2 Membership functions of the input variables "ω_e" and $\dot{\omega}_e$.

The controller gains are set as [29]:

$$\begin{aligned} &K_1^P \geq K_2^P \geq \cdots \geq K_{n-1}^P \geq K_n^P \geq 0 \\ &K_{2n-1}^P \geq K_{2n-2}^P \geq \cdots \geq K_{n+1}^P \geq K_n^P \geq 0 \\ &K_1^I \geq K_2^I \geq \cdots \geq K_{n-1}^I \geq K_n^I \geq 0 \\ &K_{2n-1}^I \geq K_{2n-2}^I \geq \cdots \geq K_{n+1}^I \geq K_n^I \geq 0 \\ &0 \leq K_1^D \leq K_2^D \leq \cdots \leq K_{n-1}^D \leq K_n^D \\ &0 \leq K_{2n-1}^D \leq K_{2n-2}^D \leq \cdots \leq K_{n+1}^D \leq K_n^D \end{aligned} \tag{10}$$

Using the typical fuzzy inference approach, the PID control input can be obtained as:

$$i_{qsd} = -\sum_{i}^{r} h_i(\omega_e) \begin{bmatrix} K_i^P(\omega_e) + K_i^I \int_0^t \omega_e d\tau \\ +K_i^D \dot{\omega}_e \end{bmatrix} \tag{11}$$

where $h_i = m_i / \sum_{j=1}^{r} m_j$ is the normalized weight of each IF-THEN rule, and it states $h_i \geq 0$, $\sum_i^r h_i = 1$. Fig. 3 shows a graphic interpretation of the fuzzy logic method to obtain the PID control input.

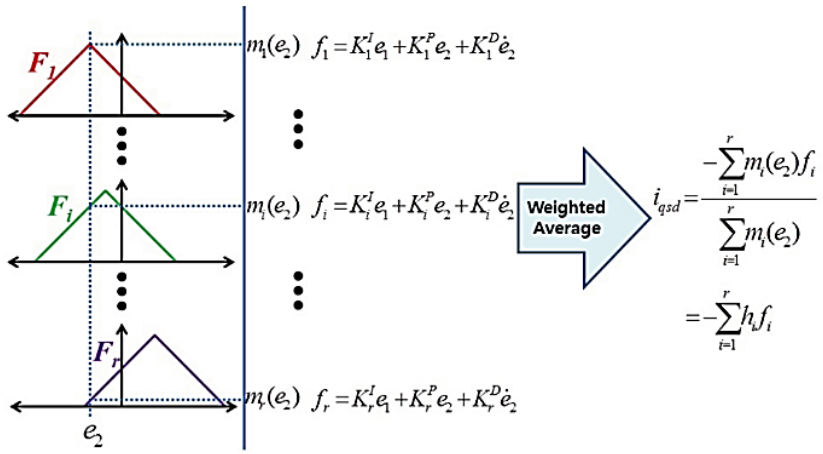


Fig. 3 Flowchart representation of the FIS [26].

According to Fig. 3, the input premise variables are the speed error and its derivative. Hence, the number of initial fuzzy sets for input is equal to 2, and only one output variable i_{qsd} is utilized. Thus, the proposed approach is more straightforward and easier than the previous heuristics-based fuzzy control techniques, such as [34]. It should be noted that if the controller gains are set:

$$\begin{aligned} &K_1^P = K_2^P = \cdots = K_{2n-1}^P = K^P > 0 \\ &K_1^I = K_2^I = \cdots = K_{2n-1}^I = K^I > 0 \\ &K_1^D = K_2^D = \cdots = K_{2n-1}^D = K^D > 0 \end{aligned} \tag{12}$$

The fuzzy control law can be changed into the conventional PID control law:

$$i_{qsd} = -(K^P \omega_e + K^I \int_0^t \omega_e d\tau + K^D \dot{\omega}_e) \tag{13}$$

Using the error vector $e = [e_1, e_2]^T$, $e_1 = \omega_e = \omega - \omega_d$ and $e_2 = \frac{de_1}{dt}$, the error dynamics can be gained as follows:

$$\begin{aligned} \dot{e}_1 &= e_2 \\ \dot{e}_2 &= -\sum_i^r h_i(e_1)\varphi_1[K_i^P e_1 + K_i^I \int_0^t e_2 d\tau + K_i^D \dot{e}_2] + \eta(t) \end{aligned} \tag{14}$$

where $\eta(t) = -\varphi_2 \omega(t) - \varphi_3 T_L(t)$ (see equations (7) and (8)).

4. FUZZY-PID CONTROLLER OPTIMIZED BY PSO

The constancy of PID coefficients included k_i, k_p, and k_d makes this controller operation is not appropriate in nonlinear systems. On the other hand, when some variation has happened in the system condition, the designed controller doesn't have excellent performance [27,28,32,33]. Employing the optimal PID controller is a suitable solution for dominating these conditions.

The PSO algorithm is one of the most effective methods that help shape fuzzy rules to be changed and optimized under a specific cost function. PSO algorithm is a meta-heuristic algorithm that solves the problems with the least information. The genetic algorithm can also act as an optimizer with three basic operators: crossover and mutation operators, to create a new population and one operator to distinguish between the generated population [35]. Albeit, the populations that are classified as inappropriate for survival, will be eliminated. Additionally, in terms of performance, GA has some significant weaknesses.

High running costs can be the main GA problem where we need to keep hundreds of chromosomes in memory and perform the algorithm for thousands of generations. Howbeit, it is notable that GA is a meta-heuristic algorithm, and its time required for complete running is faster and more optimal than systematic methods. There are numerous problem solutions known as particles in a space with a specific PSO algorithm position. PSO is a population-based algorithm and considerably similar to evolutionary computational techniques like GA. The PSO's system begins to work by collecting random solutions, searching for optimization, and generational updates. Unlike GA, PSO has no evolutionary operator, such as crossover and mutation. As previously mentioned, particles are potential solutions while flying through the problem space by looking for optimum particles [36,37].

The equations describing the behavior of particles in PSO are given below:

$$v^i[t+1] = w\, v^i[t] + c_1 r_1 (x^{i,best}[t] - x^i[t] + c_2 r_2 (x^{gbest}[t] - x^i[t]) \tag{15}$$

$$x^i[t+1] = x^i[t] + v^i[t+1] \tag{16}$$

In these equations, i specifies a pseudo time increment; v^i and x^i describe the speed and position of the ith particle; $x^{i,best}$ and x^{gbest} denote the best earlier position and best global position of the swarm, respectively. Besides, c_1 and c_2 assigned to the personal and collective learning indices, w demonstrates the inertia index, and r_1, r_2 are the random numbers varying between 0 and 1.

To find a solution using PSO, steps are as follows [38]:
Step 1: Select an approach to encode variables into particles.
Step 2: Initialization and start with an accidentally generated population of N_p particles.
Step 3: Evaluate the fitness for N_p population.
Step 4: Initialization the position and fitness of global and local variables and calculate the velocity vector.
Step 5: Calculate the new position that obtained the velocity vector and evaluate the fitness corresponding to the new position.
Step 6: Get the local best fitness and global best fitness and continue until stopping criteria.
Step 7: Stop the procedure when the necessary criterion is realized. Otherwise, repeat the algorithm by Step 4.

4.1. Fitness function

Several objective functions can be considered for improving the PMSM performance. In this research, the integral time absolute error (ITAE) index is utilized. ITAE is defined as follows:

$$ITAE = \int_0^T t|e(t)|\,dt \tag{17}$$

Velocity vector obtained based on the following equation:

$$\begin{aligned} population\ (i).velocity = {} & w * population(i).velocity \\ & + rand * C1 * population(i).Pbest\ population(i).position) \\ & + rand * C2 * (Gbest.position - population(i).position) \end{aligned}$$

4.2. Calculating new position

New position obtained using the following equation:

$$population(i).position = population(i).velocity + population(i).position$$

$$\begin{aligned} & population(i).position\ = \\ & \qquad min(max(population(i).position, lowerbound), upperbound) \end{aligned}$$

In this paper, the max iteration equal to 15 is considered as the maximum of iteration.

5. Simulation Results

Fig. 4 illuminates the block diagram of the suggested EA based TSFL-PID control system. An optical encoder calculates the rotor position (θ), and motor speed (ω). The switching frequency is assumed 5 *kHz*, and a space vector modulation (SVM) technique is in charge of controlling pulse-width modulation (PWM). Also, PMSM parameters are given in Table 1. The controlled system involves two control loops (See Fig. 4). The outer loop is suggested EA-based TSFL-PID controller, and the inner is a traditional PI current controller. The output generated by the considered controller (i_{qsd}) is applied to the input of PI current controller.

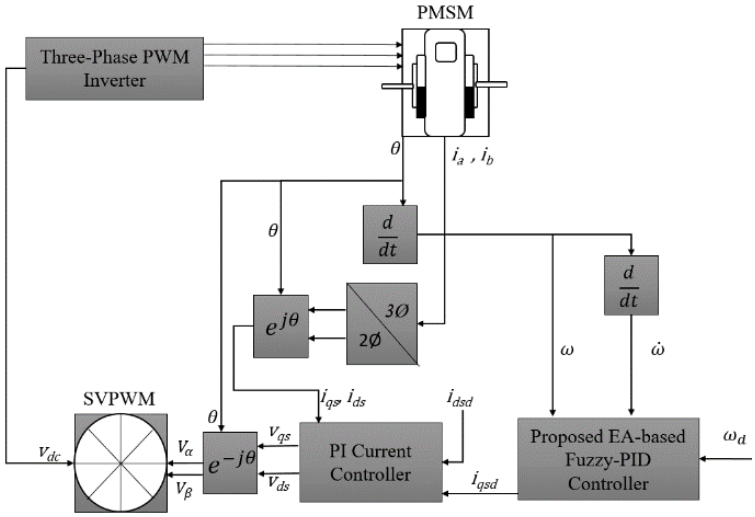


Fig. 4 Block diagram of the suggested EA based TSFL-PID controller for the PMSM system.

Table 1 PMSM Parameters

Parameter	Value
Rated output	400 W
Rated current	3.5 A
Rated speed	3000 rpm
Voltage constant (Vpeak L-L)	300 v
Number of poles (p)	4
Stator resistance (R_s)	0.18 ohm
Stator inductance ($L_s = L_q = L_d$)	0.71mh
Magnetic flux (λ_m)	0.413497
Equivalent inertia (J)	0.0008 kg.m^2
friction factor (B)	0.0001 N.m.s

In the first step, simulations are performed considering no load is affect on the PMSM. Assuming three determining speeds, including 1, 1000, and 2500 rpm, as cases a, b and c, are employed to the PMSM at start time. Membership functions are considered in the Quasi form. The fuzzy rules for designing the output based on the error and its changes are given in Table 2, in which, α is considered as P, I, D.

Table 2 Fuzzy rules for computing PID gains ($K_j^\alpha, \alpha: P, I, D$)

$\omega_e/\dot{\omega}_e$	NB	NM	ZO	PM	PB
NB	K_1^α	K_2^α	K_3^α	K_4^α	K_5^α
NM	K_6^α	K_7^α	K_8^α	K_9^α	K_{10}^α
ZO	K_{11}^α	K_{12}^α	K_{13}^α	K_{14}^α	K_{15}^α
PM	K_{16}^α	K_{17}^α	K_{18}^α	K_{19}^α	K_{20}^α
PB	K_{21}^α	K_{22}^α	K_{23}^α	K_{24}^α	K_{25}^α

With the increase in r, the control performance can be improved at the expense of the computational burden. It is well visible in the simulation results that $r = 5$ gives a satisfactory control performance. It should be noted that the proposed approach is similar to the GA-based Fuzzy-PID controller presented in [29] and simpler than the previous fuzzy methods given in [34].

Figs. 5 and 6 demonstrate the convergence diagram of the PSO and GA methods. The PSO convergency is realized faster compared to the GA. Besides, the final value of the cost function in the PSO algorithm (0.01554) is much less than GA (0.01787). Thus, the better optimal control parameters and control speed can be obtained using PSO in the current control issue. The optimal control parameters for each meta-heuristic algorithm (PSO and GA) are indicated in Table 3.

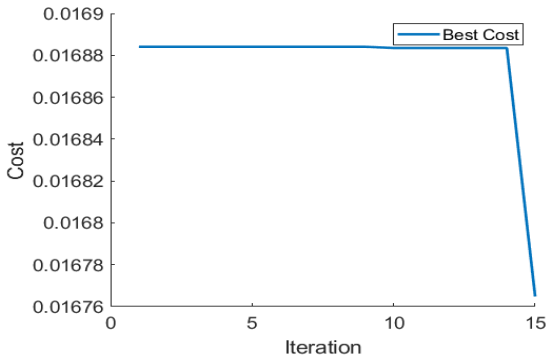


Fig. 5 The convergence of the GA algorithm

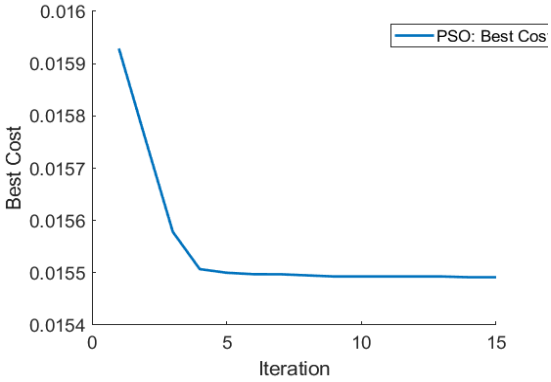


Fig. 6 The convergence of the PSO algorithm

The number of decision variables and control parameters are assumed equal for both methods. Given the following mathematical equations, the coefficients of the EA based TSFL-PID controllers for each meta-heuristic algorithm are shown in Tables 4-6. For avoiding prolongation, the first five parameters of the PID gains are mentioned in these tables.

$$
\begin{aligned}
&K_1^P = K_{2n-1}^P = \overline{K}^P, \\
&K_2^P = K_{2n-1}^P = \alpha_1 \overline{K}^P, \ldots, K_n^P = \left(\prod_{j-1}^{n-1} \alpha_j\right) \overline{K}^P \\
&K_1^I = K_{2n-1}^I = \overline{K}^I, \\
&K_2^I = K_{2n-1}^I = \beta_1 \overline{K}^I, \ldots, K_n^I = \left(\prod_{j-1}^{n-1} \beta_j\right) \overline{K}^I \\
&K_n^D = \overline{K}^D, \\
&K_{n+1}^D = K_{n-1}^D = \gamma_1 \overline{K}^D, \ldots, K_1^D = K_{2n-1}^D = \left(\prod_{j-1}^{n-1} \gamma_j\right) \overline{K}^D
\end{aligned} \tag{18}
$$

Table 3 The Achieved Optimal Control Parameters for the PSO and GA methods

Optimizers Parameters	PSO	GA
α_1	0.9814	0.8137
α_2	0.5277	0.4721
β_1	0.9116	0.8133
β_2	0.7984	0.7107
γ_1	0.7221	0.6510
γ_2	0	0.1575
δ_1	0.2319	0.6591
δ_2	0	0.3394
ε_1	0	0.8594
ε_2	0.5411	0.9733
ε_3	0.9412	0.9508

Table 4 Achieved K_j^P Parameters Related to EA based TSFL-PID controller

	GA	PSO
K_1^P	0.0400	0.04
K_2^P	0.03176	0.04
K_3^P	0.000228	0.000228
K_4^P	0.023076	0.04
K_5^P	0.040	0.04

Table 5 Achieved K_j^D Parameters Related to EA based TSFL-PID controller

	GA	PSO
K_1^D	0.00972	0
K_2^D	0.008155	0.01
K_3^D	0.01	0.01
K_4^D	0.003055	0.01
K_5^D	0.00627	0

Table 6 Achieved K_j^I Parameters Related to EA based TSFL-PID controller

	GA	PSO
K_1^I	8	8
K_2^I	5.238	8
K_3^I	2.011	8
K_4^I	5.901	8
K_5^I	8	8

Whereas the fuzzy outputs feed the PID coefficients, Takagi-Sugeno fuzzy system is used instead of the Mamdani fuzzy system structure. In the TS fuzzy structure, only the input variables have a membership function, and the output variables don't have these.

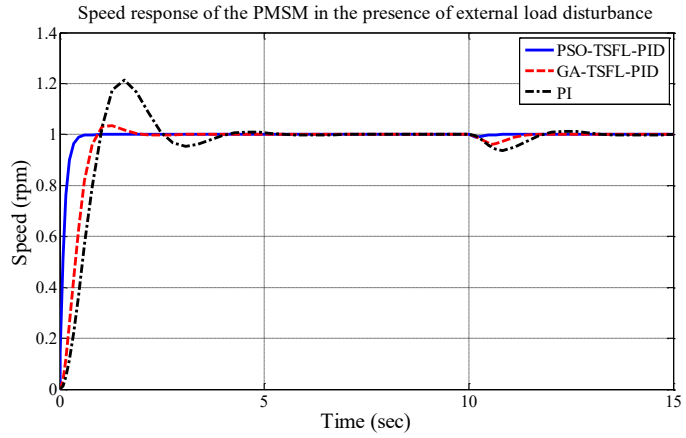
To evaluate the proposed controller's performance in speed tracking for the PMSM system, figure 7 is presented. In this figure, the PMSM responses to variation of speed and external load disturbance are demonstrated. Besides the proposed approach of this paper as the PSO based TSFL-PID, the GA based TSFL-PID and PI controllers are employed for the comparison.

Figure 7 is composed of three sub-figures. In the first sub-figure (case a), the setpoint is determined at 1 rpm. In the second and third sub-figures (cases b and c), these values are set at 1000 and 2500 rpm. Besides, for assessing the proposed method's ability in disturbance attenuation, external load disturbances have been added to the PMSM system. While a step disturbance with 0.5 rpm is applied to case 1, the amplitudes of adding the step disturbances are 200 and 1000 rpm for cases b and c, respectively. In all cases, the time of inserting external disturbances is 10 seconds.

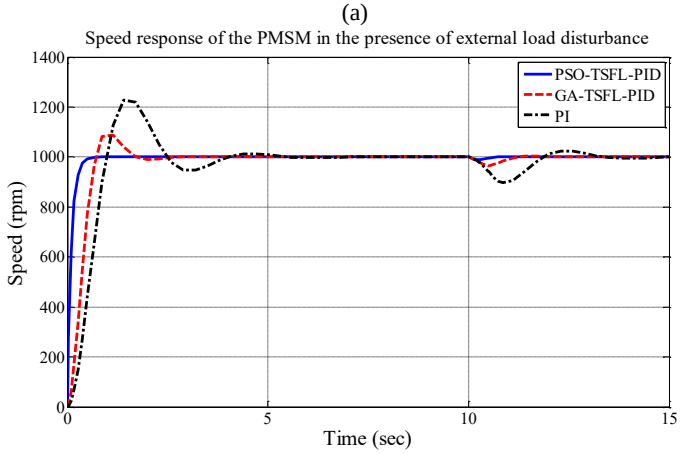
For a comprehensive comparison, the transient characteristics consist of overshoot, settling, rise, and peak times for PSO and GA based TSFL-PID and PI controllers are presented in Table 7. As can be seen, the proposed approach has the best performance in transient and settling times and overshoot for both speed tracking and disturbance attenuation. In other words, the disturbance effect on the PMSM performance is reduced via the proposed controller.

Table 7 Transient characteristic of PMSM speed control for PSO and GA Optimization Algorithm and PI Controller

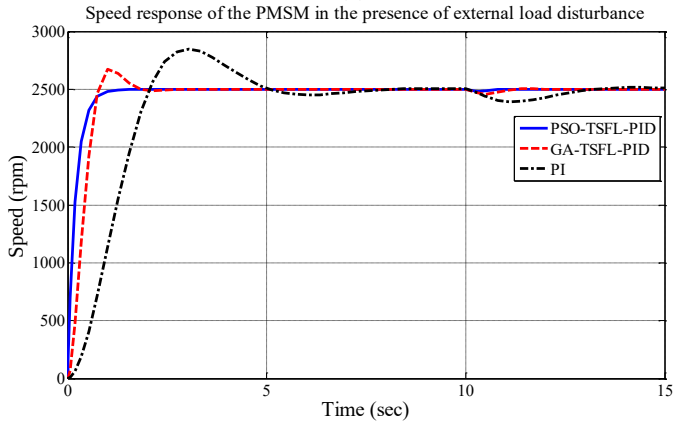
	Case 1			Case 2			Case 3		
	PSO	GA	PI	PSO	GA	PI	PSO	GA	PI
Overshoot ($M_{p\%}$)	0	3.5	21.5	0	8.75	22.7	0	6.84	14
Settling Time (t_s)	0.41	1.52	3.68	0.4	1.55	3.7	0.811	1.55	6.03
Rise Time (t_r)	0.22	0.582	0.685	0.229	0.485	0.659	0.636	0.539	1.4
Peak Time (t_p)	--	1.28	1.57	--	1.138	1.4	--	1.002	3.04



(a)



(b)



(c)

Fig. 7 Comparison between the speed response of the PMSM applying the PSO-TSFL-PID, GA-TSFL-PID, and PI controllers in three cases

6. CONCLUSION

The challenge of the speed control of the PMSM consist of nonlinear nature, cross-coupling, and air gap flux is considered in this paper. By introducing the TSFL-PID controller for this purpose, the system's overall performance is determined. Two meta-heuristic algorithms consist of the PSO and GA are individually combined by the proposed controller for the parameter's optimization. The simulation results demonstrate that although both optimization algorithms' performances are suitable in the speed control of PMSM, the PSO based TSFL-PID controller shows superiority in this particular issue. Moreover, compared to the PI controller, the best transient response in speed tracking and disturbance attenuation belongs to the proposed approach.

REFERENCES

[1] K. Zhao et al., "Sliding mode observer-based current sensor fault reconstruction and unknown load disturbance estimation for PMSM driven system", *Sensors*, vol. 17, no. 12, p. 2833, December 2017.

[2] H. Ghadiri, "Real-time Stability Assessment of Power System using ANN without Requiring Expert Experience", *Majlesi J. Electr. Eng.*, vol. 14, no. 2, pp. 43-49, June 2020.

[3] S. Heidarpoor, M. Tabatabaei and H. Khodadadi, "Speed control of a DC motor using a fractional order sliding mode controller", In Proceedings of the 2017 IEEE International Conference on Environment and Electrical Engineering and 2017 IEEE Industrial and Commercial Power Systems Europe (EEEIC/I&CPS Europe), 2017, pp. 1-4.

[4] M. Preindl, and S. Bolognani, "Model predictive direct torque control with finite control set for PMSM drive systems, part 2: Field weakening operation", *IEEE Trans. Industr. Inform.*, vol. 9, no. 2, pp. 648-657, May 2013.

[5] Y. Wang, H. Yu, Z. Che, Y. Wang and Y. Liu, "The direct speed control of PMSM based on terminal sliding mode and finite time observer", *Processes*, vol. 7, no. 9, p. 624, September 2019.

[6] K. Zhao et al., "Robust model-free nonsingular terminal sliding mode control for PMSM demagnetization fault", *IEEE Access*, vol. 7, pp. 15737-15748, February 2019.

[7] G. Liu et al., "Internal model control of permanent magnet synchronous motor using support vector machine generalized inverse", *IEEE Trans. Industr. Inform.*, vol. 9, no. 2, pp. 890-898, May 2013.

[8] Y. Zhan, J. Guan and Y. Zhao, "An adaptive second-order sliding-mode observer for permanent magnet synchronous motor with an improved phase-locked loop structure considering speed reverse", *Trans. Inst. Meas. Control*, vol. 42, no. 5, pp.1008-1021, March 2020.

[9] K. Zhou, M. Ai, D. Sun, N. Jin and X. Wu, "Field weakening operation control strategies of PMSM based on feedback linearization", *Energies*, vol. 12, no. 23, p. 4526, November 2019.

[10] F. F. El-Sousy, "Intelligent optimal recurrent wavelet Elman neural network control system for permanent-magnet synchronous motor servo drive", *IEEE Trans. Industr. Inform*, vol. 9, no. 4, pp. 1986-2003, November 2013.

[11] B. Zhang and X. Gao, "Hybrid Adaptive Integral Sliding Mode Speed Control of PMSM System Using RBF Neural Network", In Proceedings of the 2020 International Symposium on Power Electronics, Electrical Drives, Automation and Motion (SPEEDAM), 2020, pp. 17-22.

[12] F. Cao, "PID controller optimized by genetic algorithm for direct-drive servo system", *Neural Comput. Appl.*, vol. 32, no. 1, pp. 23-30, September 2020.

[13] J.-W. Jung et al., "Adaptive PID speed control design for permanent magnet synchronous motor drives", *IEEE Trans. Power Electron.*, vol. 30, no. 2, pp. 900-908, February 2014.

[14] C. Zhang et al., "Sliding observer-based demagnetisation fault-tolerant control in permanent magnet synchronous motors", *J. Eng.*, vol. 2017, no. 6, pp. 175-183, June 2017.

[15] C. Zhang et al., "Robust fault-tolerant predictive current control for permanent magnet synchronous motors considering demagnetization fault", *IEEE Trans. Ind. Electron.*, vol. 65, no. 7, pp. 5324-5334, July 2018.

[16] H. Mekki et al., "Fault tolerant design for permanent magnet synchronous motor using fuzzy speed controller", *IFAC-PapersOnLine*, vol. 49, no. 5, pp. 315-320, June 2016.

[17] L. Hongmei, C. Tao, and Y. Hongyang, "Mechanism, diagnosis and development of demagnetization fault for PMSM in electric vehicle", *Transactions of China Electrotechnical Society*, vol. 28, no. 8, pp. 276-284, August 2013.

[18] H. Li and T. Chen, "Demagnetization fault diagnosis and fault mode recognition of PMSM for EV", *Transactions of China Electrotechnical Society*, vol. 32, no. 5, pp. 1-8, March 2017.
[19] C. Xia et al., "Voltage disturbance rejection for matrix converter-based PMSM drive system using internal model control", *IEEE Trans. Ind. Electron.*, vol. 59, no. 1, pp. 361-372, January 2011.
[20] H. H. Choi and J.-W. Jung, "Discrete-time fuzzy speed regulator design for PM synchronous motor", *IEEE Trans. Ind. Electron.*, vol. 60, no. 2, pp. 600-607, February 2013.
[21] S. Chai, L. Wang, and E. Rogers, "A cascade MPC control structure for a PMSM with speed ripple minimization", *IEEE Trans. Ind. Electron.*, vol 60, no. 8, pp. 2978-2987, August 2013.
[22] I. Hassanzadeh, H. Ghadiri, and R. Dalayimilan, "Design and implemention of a simple fuzzy algorithm for obstacle avoidance navigation of a mobile robot in dynamic environment", In Proceedings of the 5th International Symposium on Mechatronics and Its Applications (ISMA), 2008, pp. 1-6.
[23] S. Wang, "Windowed least square algorithm based PMSM parameters estimation", *Math. Probl. Eng.*, vol. 2013, Article ID 131268, September 2013.
[24] Q. Xu et al., "Multiobjective optimization of PID controller of PMSM", *J. Contr. Sci. Eng.*, vol. 2014, Article ID 471609, August 2014.
[25] S. Wang, "ADRC and feedforward hybrid control system of PMSM", *Math. Probl. Eng.*,, vol. 2013, Article ID 180179, December 2013.
[26] H. H. Choi et al., "Precise PI speed control of permanent magnet synchronous motor with a simple learning feedforward compensation". *Electr. Eng.*, vol. 99, no. 1, pp. 133-139, March 2017.
[27] M. Ahmadi and H. Khodadadi, "Self-tuning PD2-PID Controller Design by Using Fuzzy Logic for Ball and Beam System", in *Fundamental Research in Electrical Engineering*, Springer, 2019 pp. 217-225.
[28] A. Dehghani and H. Khodadadi, "Designing a neuro-fuzzy PID controller based on smith predictor for heating system", In Proceedings of the 17th International Conference on Control, Automation and Systems (ICCAS). 2017, pp. 15-20.
[29] H. Ghadiri and M. R. Jahed-Motlagh, "LMI-based criterion for the robust guaranteed cost control of uncertain switched neutral systems with time-varying mixed delays and nonlinear perturbations by dynamic output feedback", *Complexity*, vol. 21, no. S2, pp. 555-578, November 2016.
[30] H. Ghadiri, M. R. Jahed-Motlagh and M. Barkhordari Yazdi, "Robust stabilization for uncertain switched neutral systems with interval time-varying mixed delays", *Nonlinear Anal.-Hybri.*, vol. 13, pp. 2-21, August 2014.
[31] H. Ghadiri, M. R. Jahed-Motlagh and M. B. Yazdi, "Robust output observer-based guaranteed cost control of a class of uncertain switched neutral systems with interval time-varying mixed delays", *Int. J. Control, Autom. Syst.*, vol. 12, no. 6, pp.1167-1179, October 2014.
[32] H. Khodadadi and H. Ghadiri, "Fuzzy Logic Self-Tuning PID Controller Design for Ball Mill Grinding Circuits Using an Improved Disturbance Observer", *Mining, Metallurgy & Exploration*, vol. 36, no. 6, pp. 1075-1090, July 2019.
[33] H. Khodadadi and A. Dehghani, "Fuzzy logic self-tuning PID controller design based on smith predictor for heating system", In Proceedings of the 16th International Conference on Control, Automation and Systems (ICCAS), 2016, pp. 161-166.
[34] Y.-S. Kung, C.-C. Huang and M.-H. Tsai, "FPGA realization of an adaptive fuzzy controller for PMLSM drive", *IEEE Trans. Ind. Electron.*, vol. 56, no. 8, pp. 2923-2932, August 2009.
[35] A. Abdollahi, A. Foruzan Tabar and H. Khodadadi, "Optimal controller design for quadrotor by genetic algorithm with the aim of optimizing the response and control input signals", *Cumhuriyet Üniversitesi Fen-Edebiyat Fakültesi Fen Bilimleri Dergisi*, vol. 36, no. 3, pp. 135-147, May 2015.
[36] H. B. Novin and H. Ghadiri, "Particle swarm optimization base explicit model predictive controller for limiting shaft torque", In Proceedings of the 5th Iranian Joint Congress on Fuzzy and Intelligent Systems (CFIS), 2017, pp. 35-40.
[37] M. Mohammadhosseini and H. Ghadiri, "A Combination of Genetic Algorithm and Particle Swarm Optimization for Power Systems Planning Subject to Energy Storage", *J. Comp. Robot.*, vol. 12, no. 1, pp. 65-76, November 2019.
[38] H. H. Choi, H. M. Yun and Y. Kim, "Implementation of Evolutionary Fuzzy PID Speed Controller for PM Synchronous Motor", *IEEE Trans. Indust. Inform.*, vol. 11, no. 2, pp. 540-547, April 2015.