State Feedback Control for a PM Hub Motor Based on Grey Wolf Optimization Algorithm

Xiaodong Sun, *Senior Member, IEEE*, Changchang Hu, Gang Lei, *Member, IEEE*,
Youguang Guo, *Senior Member, IEEE*, and Jianguo Zhu, *Senior Member, IEEE*

Abstract- **This paper presents an optimal control strategy for a permanent-magnet synchronous hub motor (PMSHM) drive using the state feedback control method plus the grey wolf optimization (GWO) algorithm. First, the linearized PMSHM mathematical model is obtained by voltage feedforward compensation. Second, to acquire satisfactory dynamics of speed response and zero *d*-axis current, the discretized state space model of the PMSHM is augmented with the integral of rotor speed error and integral of *d*-axis current error. Then, the GWO algorithm is employed to acquire the weighting matrices *Q* and *R* in liner quadratic regulator optimization process. Moreover, a penalty term is introduced to the fitness index to suppress overshoots effectively. Finally, comparisons among the GWO-based state feedback controller with and without the penalty term, the conventional state feedback controller, and the genetic algorithm enhanced PI controllers are conducted in both simulations and experiments. The comparison results show the superiority of the proposed state feedback controller with the penalty term in fast response.**

I. INTRODUCTION

With the worldwide environment deterioration, the improvement of energy efficiency has become increasingly important. Compared to traditional internal combustion engine vehicles, electrical vehicles (EVs) have higher energy efficiency and lower emissions. With the advantages of short drive chain, high dynamic performance and high efficiency, the permanent-magnet synchronous hub motors (PMSHMs) are considered alternative to the traditional permanent-magnet synchronous motor (PMSM) system with mechanical transmission for EVs [1]-[6].

The field-oriented control (FOC) system is the most commonly used control system for PMSHMs, and the cascade proportional-integral (PI) loop control structure is widely adopted. The PI controller has the advantages of simple algorithm, good robustness and high reliability. However, it needs to be tuned separately with a specified order, from the current to the speed control loops. This process can be based on analytical or experimental methods. Recently, some nature-inspired algorithms, such as genetic algorithm (GA) and particle swarm optimization (PSO) have been applied to obtain the global optimum parameters of PI controllers. And these methods can improve the performance of PI controllers [7]-[11]. However, the drawbacks of cascade PI loops still exist, e.g. the controller structure will deteriorate the system dynamics and disturbance compensation [12].

In order to improve the dynamic response and robustness of the PMSHM drive, an adaptive fuzzy neural network (AFNN) inverse control method was applied in [13]. This method contains two controllers, the AFNN controller and the inverse system controller. The PMSHM system is first decoupled by the inverse system controller, and then the AFNN controller is employed for high performance control of the pseudo-linear system. The effectiveness of this method has been proved by hardware-in-the-loop experiments. In [14], the sliding mode control method was employed to improve the robustness and static performance of hub motor drive. To suppress the fifth and seventh current harmonics, an adaptive notch filter was added to the scheme as well. However, the cascade PI loops still exist. In [15]-[17], the hub motor drive adopted direct torque control (DTC) control strategy. The DTC is well known for rapid torque response, simple structure, and less parameter dependence [15]. However, it has the shortcomings of poor low-speed performance and large torque ripple [17]. In [18], a GWO algorithm-based controller was designed for all variables, and this controller can minimize speed ripple at low-speed-high torque operations of PMSMs. It runs optimization algorithm in real time to explore the optimal control inputs that ensure satisfactory dynamics. In [19] and [20], the PSO algorithm was employed to handle nonlinear optimization in nonlinear model predictive control (NMPC-PSO). Meanwhile, since the real time implementation of optimization algorithm requires a high-performance processor, this control strategy was applied in the paralleled field-programmable gate array (FPGA) system and the satisfactory control performance can be achieved.

X. Sun and C. Hu are with the Automotive Engineering Research Institute, Jiangsu University, Zhenjiang 212013, China (email: xdsun@ujs.edu.cn, huchangchang_jd@163.com).

G. Lei and Y. Guo are with the School of Electrical and Data Engineering, University of Technology Sydney, NSW 2007, Australia (e-mail: Youguang.Guo-1@uts.edu.au, Gang.Lei@uts.edu.au).

J. Zhu is with the School of Electrical and Information Engineering, University of Sydney, NSW, 2006, Australia (e-mail: jianguo.zhu@sydney.edu.au).

The state feedback control method can be an alternative approach to combine PMSHM speed controller with intelligent algorithm. There is only one controller in this structure for all state-space variables of the PMSHM plant, and thus the drawback of the cascaded control structure can be overcome naturally. In [21], a better dynamical property, especially disturbance compensation was obtained by employing a state feedback controller (SFC). However, determining the SFC gain matrix may be a time consuming and relatively complex task. Traditionally, the SFC gain matrix is designed by using trial-and-error methods. In [22], an analytic method to determine Q and R matrices in linear quadric regulator (LQR) was proposed to obtain the SFC gain matrix. This controller presents good performance in disturbance rejection. Furthermore, in [23]-[26], the differential evolution algorithm (DEA), GA and PSO algorithms were employed to acquire the SFC gain matrix. In [24], the comparison between GA-based LQR and conventional LQR control method in doubly-fed induction generator system was presented. The comparative results show that the GA-based LQR is more stable and robust than the conventional LQR. In [25], the PSO algorithm was successfully applied to obtain the best weighting factors in quadratic cost function for a voltage-source inverter with an LC output filter. Instead of weighting matrices, only one factor is required to be set manually during the design procedure. This factor can significantly affect the closed-loop dynamics of the system, so the system characteristics can be easily designed.

In order to improve the system's dynamical response and steady-state performance, some augment variables should be introduced to the PMSHM model. The linearized state space model of the PMSHM is a type-0 MIMO system without any integral variables. To ensure zero steady-state error, some integrators need to be added to the PMSHM system. In [27], tracking errors were introduced to the state space model of permanent magnet synchronous linear motor. Compared to an SVM based direct thrust force controller (DTFC), the proposed optimal LQR based DTFC shows excellent control of flux and thrust force with faster transient response and smaller steady-state oscillations. In [21], [28] and [29], the differences of state variables were taken as model states, so the added integral variables are equal to the system outputs, and a good disturbance rejection is observed in experimental results.

However, in most of the above references with intelligent algorithms, the same fitness index (i.e., integral of speed error and d-axis current error) is employed. The fitness index is crucial to the iteration procedure and has a significant impact on the performance of the controller [30]. In some conditions, the SFC parameters obtained after multiple iterations may not be suitable for a particular situation (for example, in [24], [30] and [31], the peak overshoot occurs). In order to achieve a specific purpose (for example, no overshoots), the fitness index must be modified. Therefore, in this paper, a penalty term is added to the traditional fitness index. When overshoot occurs, it will be strictly punished by significantly increasing its fitness value. Thus, the weighting matrices that may produce overshoots will be removed in the subsequent iterations. Moreover, in this paper, the proposed SFC is based on the grey wolf optimization (GWO), which is a new member of swarm intelligence-based optimization algorithms, introduced in 2014 by Mirjalili [32]. The GWO presents the superiority in low computational complexity, high solution accuracy, convergence independence of being initial and good at dealing with local optimum, especially in latter iterations [33]-[36]. The properties mentioned above are particularly important in auto tuning procedure.

The remainder of this paper is organized as follows. In Section II, descriptions of the PMSHM system and linearization of the model are presented. In Section III, the discrete speed controller for the linearized PMSHM model is proposed. Section IV discusses the GWO algorithm and Section V introduces the application of GWO to design SFC. Section VI presents simulation results, including evaluation of fitness index and evolution trend of PMSHM speed during auto-tuning procedure with different fitness indexes. Section VII presents experimental results and comparative discussion of the proposed control approach, followed by the conclusion.

II. LINEARIZATION OF PMSHM MODEL

The nonlinear mathematical model of the surface mounted PMSHM in dq-axis reference rotor frame can be expressed as

$$\begin{cases} \dfrac{d}{dt}i_{sd} = \dfrac{1}{L_s}(-R_s i_d + u_{sd} + p\omega_m L_s i_{sq}) \\ \dfrac{d}{dt}i_{sq} = \dfrac{1}{L_s}[-R_s i_q + u_{sq} - p\omega_m(L_s i_{sd} + \psi)] \end{cases} \quad (1)$$

$$\frac{d}{dt}\omega_m = \frac{1}{J}(\frac{3}{2}p\psi i_{sq} - B_m\omega_m - T_l) \quad (2)$$

where R_s, L_s, and ψ are the resistance, inductance, and the magnetic flux of the PMSHM, respectively, u_{sq} and u_{sd} are the q-axis and d-axis voltages, respectively, ω_m is the rotor speed, p is the number of pole pairs, and B_m is the viscous friction.

In order to obtain a linearized PMSHM model, two variables are defined as

$$u_{ld} = u_{sd} + p\omega_m L_s i_q \quad (3)$$

$$u_{lq} = u_{sq} - p\omega_m(L_s i_d + \psi) \quad (4)$$

where u_{ld} and u_{lq} are the compensated voltage components. By substituting (3) and (4) into (1), the following expression can be obtained.

$$\begin{cases} \dfrac{d}{dt}i_{sd} = \dfrac{1}{L_s}(-R_s i_d + u_{ld}) \\ \dfrac{d}{dt}i_{sq} = \dfrac{1}{L_s}[-R_s i_q + u_{lq}] \end{cases} \quad (5)$$

Then the model of PMSHM given by (5) and (2) can be described in a standard form of a linear state equation as follows [37]:

$$\frac{d}{dt}\boldsymbol{x}(t) = \boldsymbol{A}\boldsymbol{x}(t) + \boldsymbol{B}\boldsymbol{u}_l(t) + \boldsymbol{E}d(t) \quad (6)$$

where

$$A = \begin{bmatrix} -\dfrac{R_s}{L_s} & 0 & 0 \\ 0 & -\dfrac{R_s}{L_s} & 0 \\ 0 & \dfrac{3p\psi}{2J_m} & -\dfrac{B_m}{J_m} \end{bmatrix}, \quad B = \begin{bmatrix} \dfrac{1}{L_s} & 0 \\ 0 & \dfrac{1}{L_s} \\ 0 & 0 \end{bmatrix}, \quad E = \begin{bmatrix} 0 \\ 0 \\ -\dfrac{1}{J_m} \end{bmatrix},$$

$$x(t) = \begin{bmatrix} i_{sd}(t) \\ i_{sq}(t) \\ \omega_m(t) \end{bmatrix}, \quad u_l(t) = \begin{bmatrix} u_{ld} \\ u_{lq} \end{bmatrix}, \quad d(t) = T_l(t)$$

III. SPEED CONTROLLER DESIGN

In order to ensure null steady-state error, integrals need to be introduced to the model. Two terms are added into the state model to assure good speed tracking and zero d-axis current in various load conditions.

$$\bar{x}(t) = \begin{bmatrix} i_{sd}(t) & i_{sq}(t) & \omega_m(t) & e_\omega(t) & e_{id}(t) \end{bmatrix}^{\mathrm{T}} \quad (7)$$

where e_{id} and e_ω are integral errors in states of i_d and ω_m, respectively.

$$e_{id} = \int (i_d - i_d^{ref})dt \quad (8)$$

$$e_\omega = \int (\omega_m - \omega^{ref})dt \quad (9)$$

The augmented state-space model with integrals can be expressed as

$$\frac{d}{dt}\bar{x}(t) = \bar{A}\bar{x}(t) + \bar{B}u_l(t) + fr(t) \quad (10)$$

where

$$\bar{A} = \begin{bmatrix} -\dfrac{R_s}{L_s} & 0 & 0 & 0 & 0 \\ 0 & -\dfrac{R_s}{L_s} & 0 & 0 & 0 \\ 0 & \dfrac{3p\psi}{2J_m} & -\dfrac{B_m}{J_m} & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \end{bmatrix} \quad \bar{B} = \begin{bmatrix} \dfrac{1}{L_s} & 0 \\ 0 & \dfrac{1}{L_s} \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \quad f = -1 \text{ and}$$

$$r(t) = \begin{bmatrix} 0 & 0 & 0 & \omega_{ref} & i_{dref} \end{bmatrix}^{\mathrm{T}}$$

The control law of this continuous model can be expressed as

$$u(t) = -K_c\bar{x}(t) = -K_{cx}x(t) - K_{ce}x_e(t) \quad (11)$$

where K_c is the constant gain matrix

$$K_c = [K_{cx} \ K_{ce}] = \begin{bmatrix} K_{cx1} & K_{cx2} & K_{cx3} & K_{ce1} & K_{ce2} \\ K_{cx4} & K_{cx5} & K_{cx6} & K_{ce3} & K_{ce4} \end{bmatrix}$$

$\bar{x}(t)$ is the augmented state vector, and $x_e(t) = [e_{id}, e_\omega]^T$ is the integral state vector.

In order to implement this speed controller in dSPACE platform, a discrete form of control law is demanded,

$$u(n) = -K_d\bar{x}(n) = -K_{dx}x(n) - K_{de}x_e(n) \quad (12)$$

where

$$K_d = [K_{dx} \ K_{de}] = \begin{bmatrix} K_{dx1} & K_{dx2} & K_{dx3} & K_{de1} & K_{de2} \\ K_{dx4} & K_{dx5} & K_{dx6} & K_{de3} & K_{de4} \end{bmatrix}$$

The gain matrix K_{de} corresponds to the integral variables e_{id} and e_ω. The discrete form of integrals can be obtained by using the back Euler integration algorithm:

$$e_{id}(n) = e_{id}(n-1) + Ts[i_d(n) - i_d^{ref}(n)] \quad (13)$$

$$e_w(n) = e_w(n-1) + Ts[\omega_m(n) - \omega^{ref}(n)] \quad (14)$$

where Ts is the sampling period, and n is discrete sample time index. The linear quadratic optimization method is used to choose the value of gain matrix. And the control law minimizes the following discrete performance index:

$$J = \sum_{n=0}^{\infty} (x_i^T(n)Qx_i(n) + u_i^T(n)Ru_i(n)) \quad (15)$$

with

$$Q = diag([q_1 \ \ q_2 \ \ q_3 \ \ q_4 \ \ q_5]), \ R = diag([r_1 \ \ r_2])$$

The weighting matrices Q and R are constant symmetric positive definite matrices, and their values have a great influence on the dynamic performance of the system. Usually, q_1, q_2 and q_3 are set to 1 as proposed in [16], [38], and [39], which means that all the state variables are considered with equal importance. Matrix R is related with control signals, and in [16] and [38], it was set to relatively high values, which gives more attention on control signal. Therefore, the overshoots can be suppressed, and the amount of energy given to the system can be reduced. The determined Q and R here are similar to that in [38], and there are

$$Q_0 = diag([1 \ \ 1 \ \ 1 \ \ 1 \ \ 1]), \ R_0 = 100 \times diag([1 \ \ 1])$$

However, for PMSHM drives, the weight matrices determined above may not be optimal, e.g. the importance of all state variables may not be equal, as well as the control signals.

In order to obtain optimal matrices, the GWO algorithm is applied in this paper.

IV. GREY WOLF ALGORITHM

The GWO is a new member of swarm intelligence-based optimization algorithms with many merits, which make it suitable for SFC optimization. Inspired by wolves' hunting, it mimics the process of tracking, encirclement, and targeting. The grey wolf society has a strict hierarchical system, from high to low, which can be divided into the leader wolf, alpha, the deputy wolf, beta, the subordinate wolf, delta, and the bottom wolf, theta. Their levels are based on their fitness values.
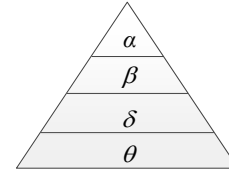

Fig. 1. Social hierarchy of grey wolves.

Fig. 1 shows the social hierarchy of grey wolves. Alpha (α) is the leader of the whole group, which gives orders to the following three levels of wolves. Beta (β) can offer advice to alpha and orders the lower ranked wolves. The responsibility

of delta (δ) is to make decisions and implement strategies. Theta (θ) represents the rest of wolves, obey orders and take actions. The main phases of grey wolf hunting are as follows:

1. Tracking and chasing the prey,
2. Pursuing and encircling the prey, and
3. Attacking the prey.

Before attacking, the wolves will encircle the prey firstly. The equations are developed to describe this behavior.

$$\vec{D} = |\vec{C} \cdot \vec{X}_P(t) - \vec{X}(t)| \tag{16}$$

$$\vec{X}(t+1) = |\vec{X}_P(t) - \vec{H} \cdot \vec{D}| \tag{17}$$

where t stands for the current iteration number, $\vec{C}$ and $\vec{H}$ are the coefficient vectors, respectively, $\vec{X}_p$ stands for the position vector of the prey, and $\vec{X}$ stands for the position vector of the grey wolf. The vectors $\vec{C}$ and $\vec{H}$ can be calculated as follows:

$$\vec{H} = 2 \cdot \vec{a} \cdot \vec{r_1} - \vec{a} \tag{18}$$

$$\vec{C} = 2 \cdot \vec{r_2} \tag{19}$$

where the value of $\vec{\alpha}$ is linearly decreased from 2 to 0 over the course of iterations, and $\vec{r_1}$ and $\vec{r_2}$ are random vectors in [0, 1].

Fig. 2 shows the grey wolf's location update. As shown, the three wolves closest to the target are named as alpha, beta and delta, respectively, and their positions need to be located firstly. The attack action is guided by them, and the other search agents are obliged to move according to their guidance.

$$\begin{cases} \vec{D}_\alpha = |\vec{C_1}\vec{X}_\alpha - \vec{X}| \\ \vec{D}_\beta = |\vec{C_2}\vec{X}_\beta - \vec{X}| \\ \vec{D}_\delta = |\vec{C_3}\vec{X}_\delta - \vec{X}| \end{cases} \tag{20}$$

$$\begin{cases} \vec{X_1} = \vec{X}_\alpha - \vec{H_1}\vec{D}_\alpha \\ \vec{X_2} = \vec{X}_\beta - \vec{H_2}\vec{D}_\beta \\ \vec{X_3} = \vec{X}_\delta - \vec{H_3}\vec{D}_\delta \end{cases} \tag{21}$$

$$\vec{X}(t+1) = \frac{\vec{X_1} + \vec{X_2} + \vec{X_3}}{3} \tag{22}$$
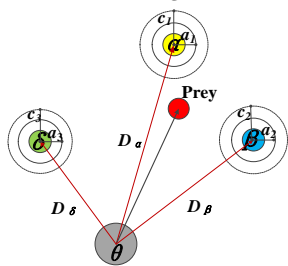

Fig. 2. Grey wolf's location update.

where $\vec{D}_\alpha$, $\vec{D}_\beta$, and $\vec{D}_\delta$ are their distance vectors with θ, respectively. $\vec{X_1}$, $\vec{X_2}$, and $\vec{X_3}$ are the movement instructions given by alpha, beta and delta, respectively.

When the value of $\vec{H}$ is greater than 1 or less than -1, the other wolves will move away from the known prey to find a more suitable target (exploration), which can allow GWO

algorithm to search globally. Moreover, the values of $\vec{C}$ are randomly distributed between 0 and 2, which will enhance exploration throughout the whole process. It can effectively avoid local optimum stagnation, especially during the final iterations.

V. APPLICATION OF GWO FOR AUTO-TUNING OF SFC

To select weighting matrices automatically with the GWO, the objective function should firstly be determined. And the proposed function should be compatible with the control objectives. It can be divided into the major objective and the secondary objectives according to their priorities. The major objective is to achieve satisfactory dynamic of angular velocity in various load torque conditions, and the secondary control objectives are: 1) zero d-axis current and 2) zero overshoot. Based on the control objectives discussed above, the proposed objective function can be expressed as

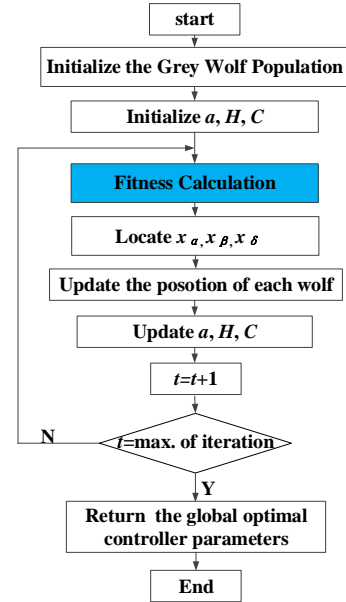$$F_1 = \frac{1}{N}\sum_{n=0}^{N}[|\Delta e_\omega(n)|nTs + |\Delta e_{id}(n)|nTs)] \tag{23}$$
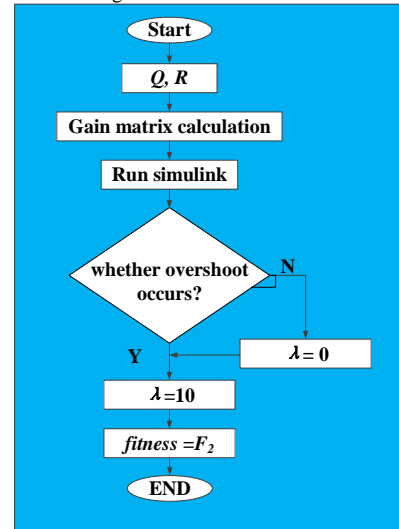

Fig. 3. Flowchart of GWO algorithm.

Fig. 4. Fitness calculation block with F_2.

In order to suppress overshoot more efficiently, the penalty term is added to the function:

$$F_2 = \frac{1}{N}\sum_{n=0}^{N}[|\Delta e_\omega(n)|\,nTs+|\Delta e_{id}(n)|\,nTs+\lambda\Delta e_\omega(n)nTs] \quad (24)$$

with

$$\Delta e_\omega(n)=\omega_m(n)-\omega^{ref}(n),\ \Delta e_{id}(n)=i_d(n)-i_d^{ref}(n),$$

$$\lambda=\begin{cases}0 & \Delta e_\omega(n)<0\\10 & \Delta e_\omega(n)>0\end{cases}$$

where $\omega^{ref}(n)$ is the speed reference and i_d^{ref} is the reference current in d-axis. The first part of the formula is related to the main control objective, and the rest of the formula is responsible for the secondary control objectives. To avoid overshoot, the penalty control is adopted. That is, if overshoot occurs, it is taken as the highest priority indicator. The coefficient λ changes to 10 from 0, and this value is selected in Section VI. The effect of the added penalty coefficient will be explored in the following part.

Fig. 3 shows the procedure of applying the GWO to find optimal controller parameters. Fig. 4 shows the procedures of fitness calculation in GWO iteration when using (24). There are four main steps.

Step 1: Generate weighting matrices Q and R randomly.

Step 2: Use the Matlab *lqr* function to obtain the gain matrices K_d.

Step 3: Substitute the value of the gain matrix to Simulink model, and simulate the motor drive to gain data required for the evaluation of the objective function.

Step 4: Calculate the fitness.

VI. SIMULATION RESULTS

In this section, in order to evaluate the performance of the proposed SFC, comparisons have been carried out by using the MATLAB/Simulink. To ensure the safe and proper operation of electrical drive, the q-axis current is constrained by $|i_q|<10$ A. The PMSHM parameters are listed in Table I.

TABLE I
SELECTED PARAMETERS OF THE PMSHM DRIVE

Parameter	Symbol	Value	Unit
Stator resistance	R_s	0.8	Ω
Stator inductance	Ls	4.5	mh
No. of pole pairs	P	22	
Magnet flux	Ψ	0.215	wb
Inertia	I_m	0.03	kgm²
Frictional coefficient	B_m	0.0006	Nm/s
Rated speed	N	360	rpm
Rated power	P_N	3000	w
DC-link voltage	U_{dc}	420	v
Sampling period	Ts	1e-5	s
Electrical time constant	τ_e	5.625	ms
Mechanical time constant	τ_m	4.6	ms

As the most widely used PMSHM control method, the conventional FOC method contains 3 PI controllers, one for speed loop and two for current loop. In order to make a fairly comparison between SFCs and the conventional FOC, the parameter determination of the PI controllers refers to the method of using genetic algorithm (GA) in [7]. And the same fitness index F_1 is employed in PI controllers' parameters tuning procedure. Table II lists the selected PI parameters.

TABLE II
PARAMETERS OF THE PI CONTROLLERS

Controller	P	I
Speed controller	0.13	6.31
q-axis current controller	5.01	76.72
d-axis current controller	4.34	83.57

To test and verify the SFC on the dSPACE platform, the discrete mode is simulated. The block diagram of the proposed drive with SFC is shown in Fig. 5. K_d is the gain matrix of SFC. After the weighting matrices Q and R are selected, it can be calculated in MATLAB by using the following formula:

$$[K_d,\sim,\sim]=lqr(A,B,Q,R) \quad (25)$$

Control signal $u_s=[u_{sd}\ u_{sq}]^T$ can be calculated from (3) and (4)
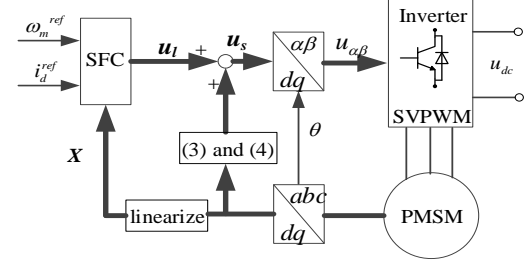

Fig. 5. Block diagram of PMSHM drive with proposed SFC.

Before starting the optimization procedure, some parameters of GWO should be selected firstly. The size of wolf population will influence the time and accuracy of the optimization directly. More wolves will consume more time, but too few wolves may cause failure in finding the optimal gain matrix. The number of wolves and maximum iterations are selected on the basis of information contained in [30] and [32] In order to ensure that the gain matrix K_d can be selected within a large range of values, the upper bounds u_b and lower bounds l_b are selected as 1×10^6 and 1×10^3, respectively.

Multiple simulations with different λ values are performed to select an appropriate value for the penalty coefficient. In order to facilitate the comparison of the effects with different λ values on suppressing overshoot, the overshoot error cumulative value (*OECV*) after 40 iterations with GWO is defined as a comparison criterion.

$$OECV=\sum_{n_1}^{n_2}e_\omega(n) \quad (26)$$

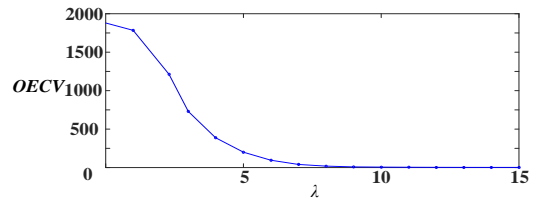where n_1 and n_2 are the moment indexes when the overshoot starts and ends, respectively.

Fig. 6. *OECV* with different λ.

The values of *OECV* with λ from 0 to 15 are shown in Fig.6. As can be observed, when λ is greater than 9, the speed overshoot can be effectively suppressed. Therefore, $\lambda=10$ is taken in this paper. Selected parameters of GWO are recorded in Table III.

In order to highlight the advantages of GWO in computational efficiency and achieving a global optimum, two other evolutionary algorithms, PSO and GA are also adopted for parameter tuning process with objective function (23). These three evolutionary algorithms are examined in MATLAB R2016a environment, on a PC with i5-4200 CPU @ 2.5 GHz with 8GB RAM. The overall trends of F_1 during parameter tuning process, as well as the total time needed to complete 40 iterations with PSO, GWO and GA, are shown in Figs.7 (a)-(c), respectively. The evolution of F_2 with GWO is recorded in Fig.7 (d).
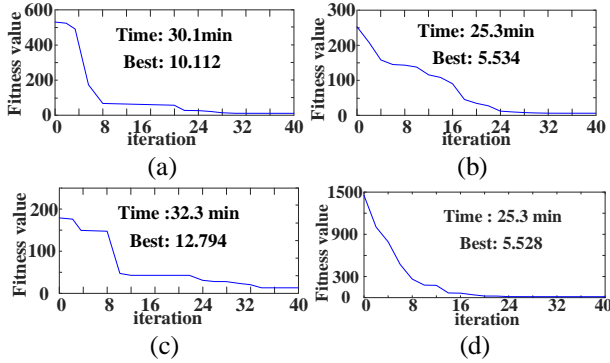

Fig. 7. Evolution of fitness value during auto-tuning procedure: (a) PSO with F_1, (b) GWO with F_1, (c) GA with F_1, and (d) GWO with F_2.

Comparing Fig.7 (b) with Fig.7 (a) and Fig.7 (c), it can be found that with the same objective function, GWO takes 25.3 mins to complete 40 iterations while the PSO takes 30.1 mins and GA takes 32.3 mins. Considering that most of the time is spent on simulation performed in SIMULINK, GWO shows a much better computational efficiency compared to the other two. Moreover, as can be observed from Fig. 7, with the same objective function, the best F_1 after 40 iterations with PSO is 10.112, with GA is 12.794, while GWO finds the smallest F_1 5.534. This means that the parameter tuning procedure with PSO and GA are more likely trapped in a local optimum, but GWO has managed to avoid these traps and find the better results. These comparisons prove that the GWO is more suitable for parameter tuning of SFC.
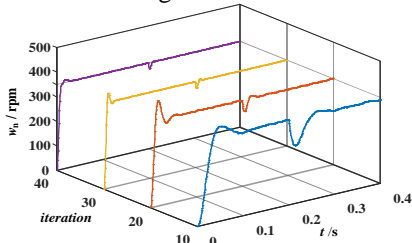

Fig. 8. Evolution of speed during auto-tuning procedure with GWO and F_1
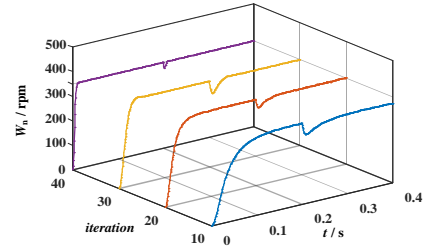

Fig. 9. Evolution of speed during auto-tuning procedure with GWO and F_2

It should be noted that, due to the introduction of penalty term, the initial value of F_2 is very large (as in Fig.7 (d)), but it decreases rapidly in following iterations. When it has iterated for 24 times, the difference between F_2 and F_1 with GWO has become very small, and after 40 iterations, F_2 and F_1 are approximately the same. The best fitness indexes after 40 iterations is $F_2=5.528$. Figs. 8 and 9 show the evolution of speed during the auto tuning procedure with GWO ($\omega_n^{ref} = 350$ rpm, $T_l = 10$ Nm at $t = 0.2$ s).

Moreover, it is worthy to point out that the importance of state tracking error and control energy loss in PMSHM control process are represented by matrices Q and R, respectively. The main diagonal elements of the matrix Q represent the relative importance of each indicator error, and the importance of control signals are compared by r_1 and r_2. The controller will give more restraints on the important variables. The final values of weighting matrices obtained after 40 iterations are as follows

$$Q_1 = \begin{bmatrix} 103.8 & 0 & 0 & 0 & 0 \\ 0 & 2.08 & 0 & 0 & 0 \\ 0 & 0 & 0.11 & 0 & 0 \\ 0 & 0 & 0 & 39.34 & 0 \\ 0 & 0 & 0 & 0 & 31.23 \end{bmatrix}, R_1 = \begin{bmatrix} 1e-3 & 0 \\ 0 & 50.1 \end{bmatrix}$$

$$Q_2 = \begin{bmatrix} 78.8 & 0 & 0 & 0 & 0 \\ 0 & 1.26 & 0 & 0 & 0 \\ 0 & 0 & 0.09 & 0 & 0 \\ 0 & 0 & 0 & 62.56 & 0 \\ 0 & 0 & 0 & 0 & 10.11 \end{bmatrix}, R_2 = \begin{bmatrix} 1e-3 & 0 \\ 0 & 48.43 \end{bmatrix}$$

The corresponding gain matrixes are:

$$K_1 = \begin{bmatrix} 322.1 & 0 & 0 & 0 & 176.72 \\ 0 & 1.49 & 0.0523 & 0.8861 & 0 \end{bmatrix},$$

$$K_2 = \begin{bmatrix} 280.64 & 0 & 0 & 0 & 100.55 \\ 0 & 1.46 & 0.05 & 1.14 & 0 \end{bmatrix}$$

The performances of the PMSHM drive are investigated at different operating conditions. The comparisons are made among the traditional SFC (SFC0), the SFC tuned by GWO with objective function F_1 (SFC1), the SFC tuned by GWO with objective function F_2 (SFC2), and the PI controller tuned by GA (GA-PI). Figs. 10-12 illustrate the dynamic behaviors of SFC0, SFC1, SFC2 and GA-PI at no load, constant load and variable load conditions, respectively. Since the high

efficiency of the PMSHM happens during 300-400 rpm, we set ω_n^{ref} = 350 rpm for all conditions.

At no load condition

When load torque is set to zero, all the motor drives can track reference speed without steady-state error. As shown in Fig. 10 (a), the drive with SFC0 has a relatively long rise time, while SFC1, SFC2 and GA-PI based drives can reduce it greatly, but fairly obvious overshoots can be observed in GA-PI and SFC1 cases. As for the SFC2 based drive, thanks to the introduction of penalty term λ in (24), it allows tracking reference speed almost at the same efficiency as SFC1 but without any overshoots. Fig. 10 (b) shows the measured i_d for SFC0, SFC1, SFC2 and GA-PI. The reference value of i_d is set to zero in this test. Helped by the second part in fitness index, SFC1 and SFC2 have smaller oscillation in i_d than SFC0. i_q of the controllers is recorded in Fig.10 (c). As can be seen that, i_q reaches its upper limit both in SFC1 and GA-PI, while in SFC0 and SFC2, the peak values of i_q are smaller.
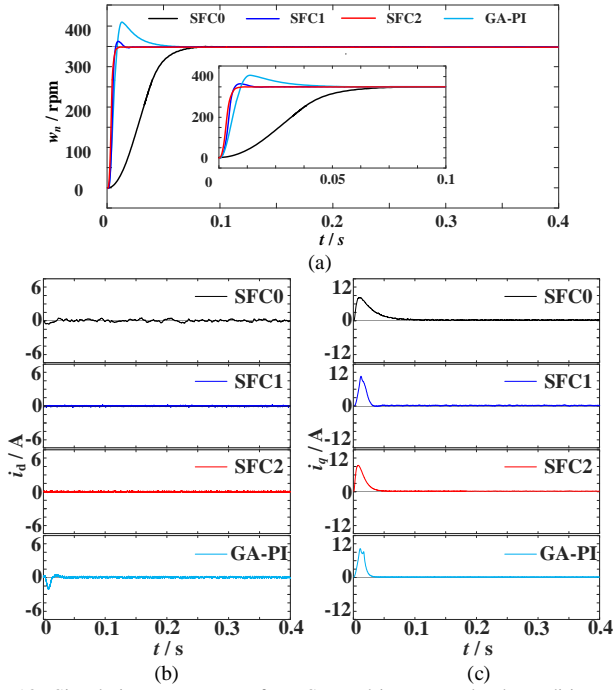

Fig. 10. Simulation responses of PMSHM drive at no load conditions: (a) speed, (b) d-axis current, and (c) q-axis current.

At constant load condition

In this condition, the load torque is set to 10 Nm in the whole period of simulation. The responses of the speed and current are recorded in Fig. 11. Compared to the no load condition, the rise time with SFC0 and GA-PI become obviously longer, and it seems very hard for the SF0 to reach the reference speed. This condition may be caused by a relatively small value of q_4 in Q_0. As mentioned above, larger value in weighting matrix means more restraints on the corresponding variable, and q_4 is related to speed error integral. In Q_1, q_4 = 39.34, and in Q_2, q_4 = 62.56, but in Q_0, q_4 = 1. That means that SFC1 and SFC2 give much more attention on speed reference tracking than SFC0. Thus, the superiority of

SFC1 and SFC2 in steady state error elimination is obvious. Compared to SFC0, SFC1, SFC2 and GA-PI have reduced the rise time of PMSHM speed by 82.4%, 79.4% and 54.1%, respectively. As recorded in Fig. 11(c), i_q becomes larger in this condition to overcome the constant load torque. This causes a more serious integral windup condition in SFC1 and GA-PI. Thus, the duration of speed overshoot with GA-PI and SFC1 has increased by 20.5% and 39.8%, respectively. Also, the amplitude of speed overshoot with GA-PI and SFC1 are 3.4% and 15.5%, while it is 0 for SFC2.
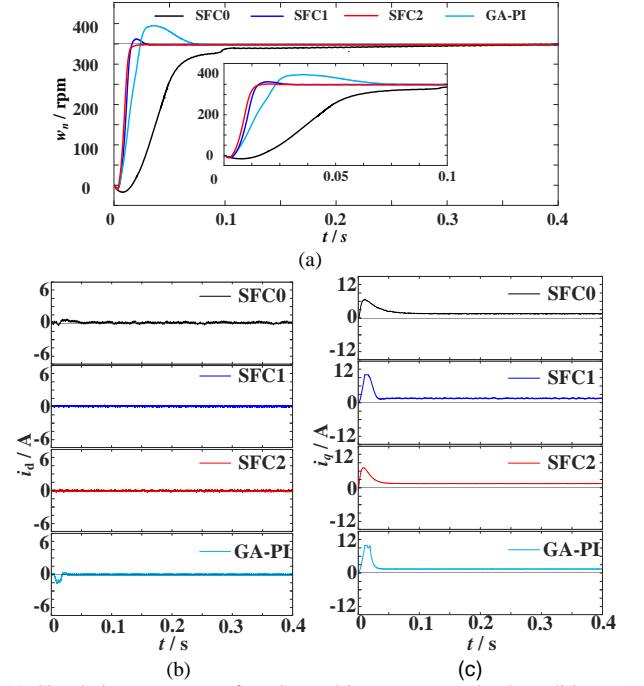

Fig. 11. Simulation responses of PMSHM drive at constant load conditions: (a) speed, (b) d-axis current, and (c) q-axis current.
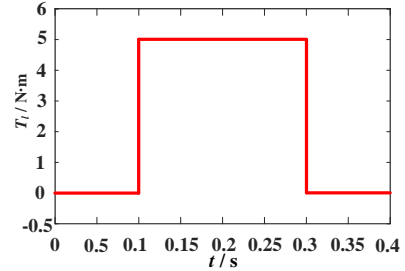

Fig. 12. Load torque profile.

At variable load condition

In this condition, the load torque is initially set to zero. When the PMSHM operates stably at speed of 350 rpm, the load torque used for simulations is applied to test the disturbance rejection property, as shown in Fig. 12. The responses of the speed and current are shown in Fig. 13.

As shown in Fig.13, when the load torque is applied, the three SFC drives have experienced almost the same speed drop, and their amplitudes are smaller than that of GA-PI. However, the times they spend to regain stability are quite different. It takes only a very short period for SFC1 and SFC2

to resettle down, but SFC0 does not make its speed back to the reference value until the load is removed. Similarly, when the load is removed, the three SFC drives produce the similar speed overshoot, but both SFC1 and SFC2 recover much more quickly than SFC0 does. This may be mainly explained by their difference in R, the weighting matrix related with control signals. Elements in R_0 are all greater than those in R_1 and R_2. Therefore, SFC0 will give more constraints on the control signal, e.g. the amount of energy given to the system will be more limited. Thus, the SFC1 and SFC2 have a better disturbance rejection property than GA-PI, while this capacity of SFC0 is unsatisfactory. Moreover, when compared with SFC1, the advantage of SFC2 in suppressing overshoots and disturbance rejection is exhibited.
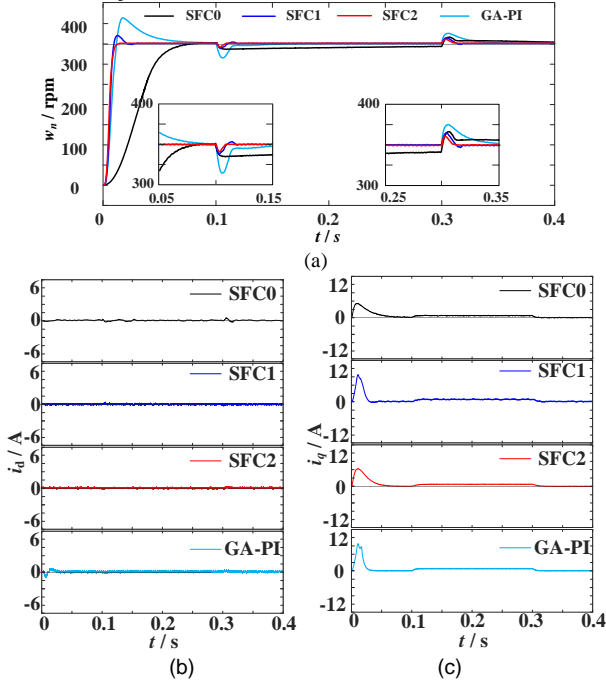

Fig. 13. Simulation responses of PMSHM drive at variable load conditions: (a) speed, (b) d-axis current, and (c) q-axis current.

In the above simulation cases, the proposed SFC2 performs the best in terms of reference speed tracking, load compensation and disturbance rejection.

VII. EXPERIMENTAL RESULTS

To validate the system performance with the proposed controller, several experiments have been carried out. The experimental setup is shown in Fig. 14. The test bench consists of a PMSHM, a torque sensor, and a magnetic powder brake.

The proposed control scheme is implemented in a dSPACE DS1401 PPC. For comparative purposes, the experiments use the same values of reference speed and load torque as the simulations.

The experimental responses of the PMSHM with different controllers under no load, constant load and variable load conditions are presented in Figs.15-17, respectively. It should be noted that the speed responses of the PMSHM coincide well with the simulation results shown in Figs. 11-13. As shown in Figs. 15-17, steady-state error-free operation can be obtained with SFC1, SFC2 and GA-PI in all conditions. While for SFC0, due to the improper selection of weighting matrices, it shows unsatisfactory reference tracking performance in some cases. Moreover, with SFC1 and SFC2 controllers, better compensation of external load torque is observed. d-axis current is kept to zero at steady state with all controllers, but the noises of SFC0, SFC1, SFC2 and GA-PI controllers are different. The SFC1 and SFC2 controllers have obvious advantages in reducing current oscillation. In addition, for SFC1 and SFC2 controllers, the noise in d-axis current is mainly affected by the value of q_1 obtained from auto-tuning process, and greater q_1 means more emphasis imposed on i_d by the controller. Compared with $q_1=1$ in SFC0 case, the q_1 selected after 40 GWO iterations with F1 and F2 are 103.8 and 78.8 respectively. Moreover, compared with SFC1 controller, the speed drop and overshoot with SFC2 controller, are slightly smaller when the load is imposed and removed, owing to the added penalty term in fitness index. Also, compared with GA-PI, the oscillation amplitude and saturation time of i_q are obviously smaller in SFC1 and SFC2.
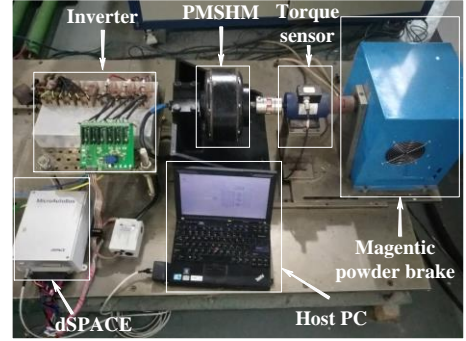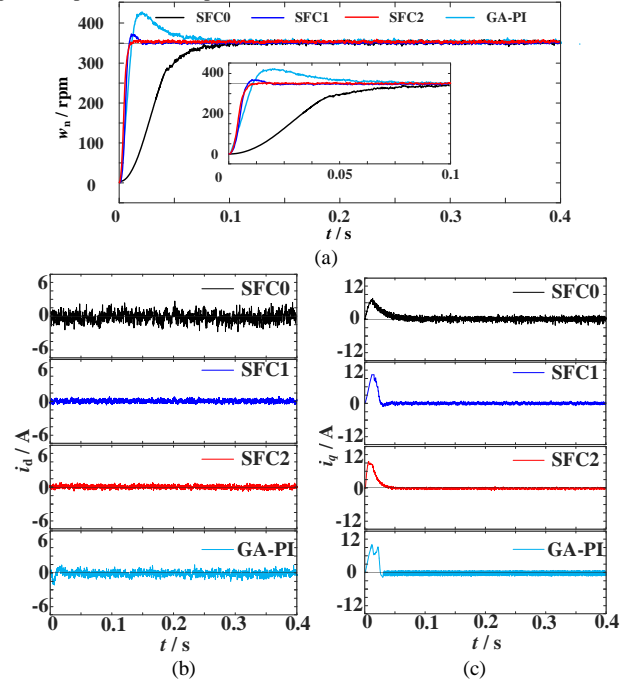

Fig. 14. Experimental setup.

Fig. 15. Experimental responses of PMSHM drive at no load conditions: (a) speed, (b) d-axis current, and (c) q-axis current.
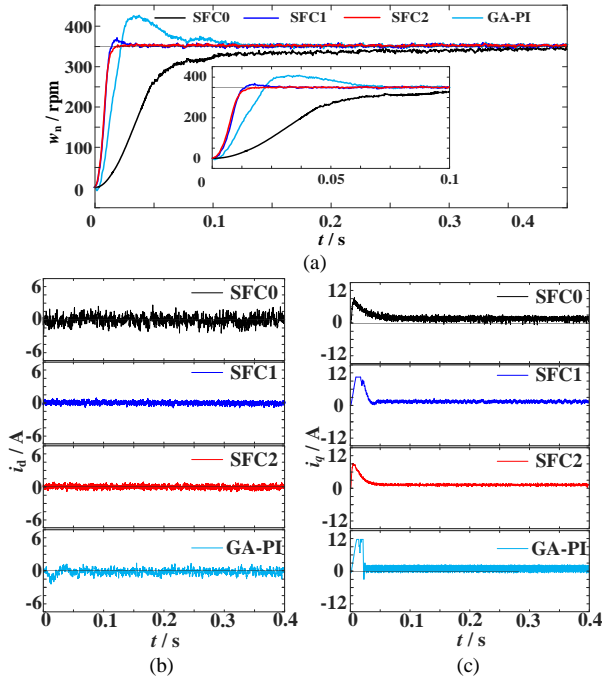
Fig. 16. Experimental responses of PMSHM drive at constant load conditions: (a) speed, (b) *d*-axis current, and (c) *q*-axis current.
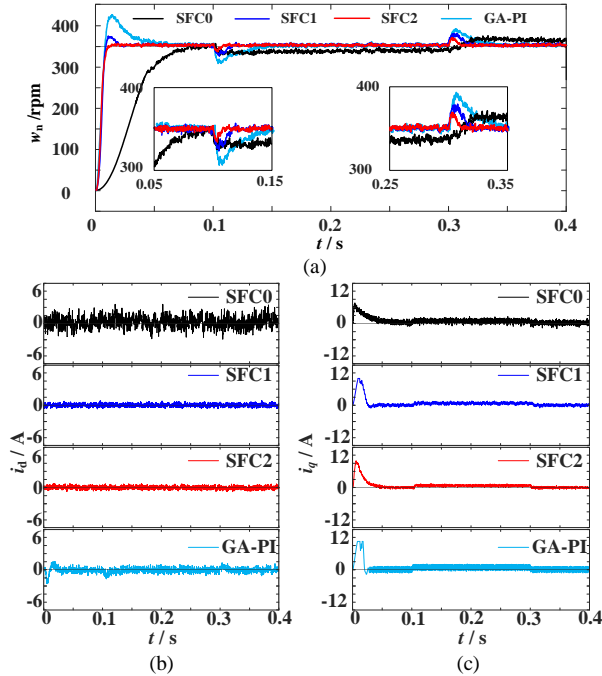

Fig. 17. Experimental responses of PMSHM drive at variable load conditions: (a) speed, (b) *d*-axis current, and (c) *q*-axis current.

TABLE IV
DYNAMIC RESPONSE OF PMSHM AT NO LOAD AND CONSTANT LOAD CONDITIONS

Controller	Load conditions	Rise time (ms)	Peak overshoot (%)	Peak time (s)	Settling Time (s)
SFC0	No load	55	-	-	0.115
	Constant load	85.2	-	-	>0.4
SFC1	No load	7.2	4	0.012	0.018
	Constant load	18	3.6	0.022	0.03
SFC2	No load	7.9	-	-	0.02
	Constant load	20	-	-	0.032
GA-PI	No load	8.2	19.8	0.014	0.085
	Constant load	22	16.5	0.032	0.14

TABLE V
DYNAMIC RESPONSE OF PMSHM AT VARIABLE LOAD CONDITIONS

| Load conditions	Motor Characteristic	SFC0	SFC1	SFC2	GA-PI
Load applied	Speed drop (%)	6.6	6.4	4.3	12.3
	Transient time (ms)	>200	19	12	53
Load removed	Peak overshoot (%)	5.2	6.7	4	10.6
	Transient time (ms)	>100	18.8	12.5	60

VIII. CONCLUSION

This paper proposed a GWO-based state feedback controller for high performance control of PMSHM drives. The proposed controller aims to ensure satisfactory dynamics of angular velocity under varying load conditions. Values of weighting matrices needed for calculation of discrete state feedback speed controllers were obtained by using GWO algorithm. Compared with GA and PSO, GWO shows obvious advantages in terms of computational efficiency and avoiding local optimization in this test. To suppress overshoot efficiently, a penalty term was introduced to the fitness index to filter out weighting matrices that will produce overshoot. Meanwhile, to validate the effectiveness of the penalty term, to check the superiority of automatic parameter selection with GWO and to make comparison with the most widely employed scheme, SFC1, SFC0 and GA-PI were selected as control groups. Tests were implemented under dSPACE 1401 control board, and the obtained results underlined the improvement of the proposed speed controller.

In addition, in Table IV, different indicators, including rise time, peak overshoot, peak time and settling time, are reported for no load and constant load conditions. In Table V, different indicators such as speed drop, peak overshoot and transient time are shown in variable load condition. As it can be appreciated, these indicators highlight the superiority of the proposed controller.

REFERENCES

[1] A. Dalal and P. Kumar, "Design, prototyping, and testing of a dual-rotor motor for electric vehicle application," *IEEE Trans. Ind. Electron*, vol. 65, no. 9, pp. 7185-7192, Sept. 2018.
[2] P. Zheng, Y. Sui, J. Zhao, C. Tong, T. Lipo, and A. Wang, "Investigation of a novel five-phase modular permanent-magnet in-wheel motor," *IEEE Trans. Magn.*, vol. 47, no. 10, pp. 4084-4087, Oct. 2011.
[3] C. Ma, *et al.*, "Sound quality evaluation of noise of hub permanent-magnet synchronous motors for electric vehicles," *IEEE Trans. Ind. Electron.*, vol. 63, no. 9, pp. 5663-5673, Sept. 2016.
[4] C. Xia, S. Wang, Z. Wang, and T. Shi, "Direct torque control for VSI–PMSMs using four-dimensional switching-table," *IEEE Trans. Power Electron.*, vol. 31, no. 8, pp. 5774–5785, Nov. 2016.

[5] L. Jarzebowicz, "Errors of a linear current approximation in high-speed PMSM drives," *IEEE Trans. Power Electron.*, vol. 32, no. 11, pp. 8254–8257, Apr. 2017.

[6] I. Jeong, B. J. Hyon, and K. Nam, "Dynamic modeling and control for SPMSMs with internal turn short fault," *IEEE Trans. Power Electron.*, vol. 28, no. 7, pp. 3495–3508, Jul. 2013.

[7] M. J. Neath, A. K. Swain, U. K. Madawala, and D. J. Thrimawithana, "An optimal PID controller for a bidirectional inductive power transfer system using multiobjective genetic algorithm," *IEEE Trans. Power Electron.*, vol. 29, no. 3, pp. 1523-1531, Mar. 2014.

[8] B. L. G. Costa, V. D. Bacon, S. A. O. da Silva, and B. A. Angélico, "Tuning of a PI-MR controller based on differential evolution metaheuristic applied to the current control loop of a Shunt-APF," *IEEE Trans. Ind. Electron.*, vol. 64, no. 6, pp. 4751-4761, Jun. 2017.

[9] R. Kannan, N. Gayathri, M. Natarajan, R. S. Sankarkumar, L. V. Iyer, and N. C. Kar, "Selection of PI controller tuning parameters for speed control of PMSM using biogeography based optimization algorithm," in *Proc. IEEE Int. Conf. Power Electron., Drives Energy Syst. (PEDES)*, 2016, pp. 1-6.

[10] S. Agarwal, D. Yadav, and A. Verma, "Speed control of PMSM drive using bacterial foraging optimization," in *Proc. IEEE Uttar Pradesh Section Int. Conf. Electrical, Computer and Electronics (UPCON)*, 2017, pp. 84-90.

[11] D. Yadav and A. Verma, "Performance analysis of permanent magnet synchronous motor drive using particle swarm optimization technique," in *Proc. Int. Conf. on Emerging Trends in Electrical Electronics & Sustainable Energy Systems (ICETEESES)*, 2016, pp. 280-285.

[12] M. Preindl and S. Bolognani, "Model predictive direct speed control with finite control set of PMSM drive systems," *IEEE Trans. Power Electron.*, vol. 28, no. 2, pp. 1007-1015, Feb. 2013.

[13] Y. Li , Z. Bohan, and X. Xing, "Robust control for permanent magnet in-wheel motor in electric vehicles using adaptive fuzzy neural network with inverse system decoupling," *Transactions of the Canadian Society for Mechanical Engineering*, pp. 1-12, 2018.

[14] Y. Fan, Q. Zhang, W. Wang, and X. Zhou, "Speed regulation system of a flux-modulated permanent-magnet in-wheel motor based on sliding mode control and adaptive notch filter," *IEEE Trans. Energy Convers.*, vol. 33, no. 4, pp. 2183-2190, Dec. 2018.

[15] Y. Fan, L. Zhang, M. Cheng, and K. T. Chau, "Sensorless SVPWM-FADTC of a new flux-modulated permanent-magnet wheel motor based on a wide-speed sliding mode observer," *IEEE Trans. Ind. Electron.*, vol. 62, no. 5, pp. 3143-3151, May 2015.

[16] L. Huan and H. Zhang, "An improved DTC for in-wheel BLDC motors in micro all-electric vehicles," *Automatika*, vol. 57, no. 3, pp. 648-659, 2016.

[17] Y. Fan, L. Zhang, and M. Wei, "The improved direct torque control of a new self-decelerating permanent-magnet in-wheel motor for electric vehicles," in *Proc. IEEE Vehicle Power and Propulsion Conf. (VPPC)*, 2013, pp. 1-5.

[18] A. Djerioui, A. Houari, M. Ait-Ahmed, M. Benkhoris, A. Chouder, and M. Machmoum, "Grey wolf based control for speed ripple reduction at low speed operation of PMSM drives," *ISA Trans.*, vol. 74, pp. 111-119, Jan. 2018.

[19] O. Wallscheid, U. Ammann, and J. Boecker, "Real-time capable model predictive control of permanent magnet synchronous motors using particle swarm optimisation," in *Proc. Int. Exhibition and Conf. for Power Electronics, Intelligent Motion, Renewable Energy and Energy Management*, 2016, pp. 1-8.

[20] F. Xu, H. Chen, X. Gong, and Q. Mei, "Fast nonlinear model predictive control on FPGA using particle swarm optimization," *IEEE Trans. Ind. Electron.*, vol. 63, no. 1, pp. 310-321, Jan. 2016.

[21] C. J. Meirinho, A. Bartsch, J. de Oliveira, and M. S. M. Cavalca, "An optimal MIMO control approach for PMSM drives," in *Proc. Brazilian Power Electron. Conf. (COBEP)*, 2017, pp. 1-6.

[22] M. Brasel, "A gain-scheduled multivariable LQR controller for permanent magnet synchronous motor," in *Proc. 19th Int. Conf. Methods Models Auto. Robot. (MMAR)*, 2014, pp. 722-725.

[23] X. Deng, X. Sun, R. Liu, and W. Wei, "Optimal analysis of the weighted matrices in LQR based on the differential evolution algorithm," in *Proc. 29th Chin. Control Decision Conf. (CCDC)*, 2017, pp. 832-836.

[24] R. Bhushan, K. Chatterjee, and R. Shankar, "Comparison between GA-based LQR and conventional LQR control method of DFIG wind energy system," in *Proc. 3rd Int. Conf. on Recent Advances in Info. Technology (RAIT)*, 2016, pp. 214-219.

[25] B. Ufnalski, A. Kaszewski, and L. M. Grzesiak, "Particle swarm optimization of the multioscillatory LQR for a three-phase four-wire voltage-source inverter with an LC output filter," *IEEE Trans. Ind. Electron.*, vol. 62, no. 1, pp. 484-493, Jan. 2015.

[26] K. Paponpen and M. Konghirun, "LQR state feedback controller based on particle swarm optimization for IPMSM drive system," in *Proc. IEEE 10th Conf. Ind. Electron. Appl. (ICIEA)*, 2015, pp. 1175-1180.

[27] M. A. M. Cheema, J. E. Fletcher, M. F. Rahman, and D. Xiao, "Optimal, combined speed, and direct thrust control of linear permanent magnet synchronous motors," *IEEE Trans. Energy Convers.*, vol. 31, no. 3, pp. 947-958, Sept. 2016.

[28] P. Teppa Garran, V. Nardone, and J. Rodriguez Diez, "LQR control employing output derivative measures," *IEEE Latin America Trans.*, vol. 13, no. 8, pp. 2538-2544, Aug. 2015.

[29] T. K. Boukas and T. G. Habetler, "High-performance induction motor speed control using exact feedback linearization with state and state derivative feedback," *IEEE Trans. Power Electron.*, vol. 19, no. 4, pp. 1022-1028, July 2004.

[30] T. Tarczewski and L. M. Grzesiak, "An application of novel nature-inspired optimization algorithms to auto-tuning state feedback speed controller for PMSM," *IEEE Trans. Ind. Appl.*, vol. 54, no. 3, pp. 2913-2925, May/Jun. 2018.

[31] T. Tarczewski and L. M. Grzesiak, "Constrained state feedback speed control of PMSM based on model predictive approach," *IEEE Trans. Ind. Electron.*, vol. 63, no. 6, pp. 3867-3875, Jun. 2016.

[32] S. Mirjalili, S. Mohammad, and A. Lewis, "Grey wolf optimizer," *Advances in Engineering Software*, vol. 69, no. 3, pp. 46–61, 2014.

[33] S. Mohanty, B. Subudhi, and P. K. Ray, "A new MPPT design using grey wolf optimization technique for photovoltaic system under partial shading conditions," *IEEE Trans. Sustainable Energy*, vol. 7, no. 1, pp. 181-188, Jan. 2016.

[34] R. Precup, R. David, and E. M. Petriu, "Grey wolf optimizer algorithm-based tuning of fuzzy control systems with reduced parametric sensitivity," *IEEE Trans. Ind. Electron.*, vol. 64, no. 1, pp. 527-534, Jan. 2017.

[35] S. Sharma, S. Bhattacharjee, and A. Bhattacharya, "Grey wolf optimisation for optimal sizing of battery energy storage device to minimise operation cost of microgrid," *IET Generation, Transmission & Distribution*, vol. 10, no. 3, pp. 625-637, Feb. 2016.

[36] J. Song, F. Dong, J. Zhao, H. Wang, Z. He, and L. Wang, "An efficient multi-objective design optimization method for PMSLM based on extreme learning machine," *IEEE Trans. Ind. Electron.* doi: 10.1109/TIE.2018.2835413.

[37] A. Masmoudi, L. M. Grzesiak, and T. Tarczewski, "PMSM servo drive control system with a state feedback and a load torque feedforward compensation," *The Int. Journal for Computation and Mathematics in Electrical and Electronic Engineering*, vol. 32, no. 1, pp. 364-382, 2012.

[38] Robin and M. P. R. Prasad, "Current and speed control of a permanent magnet synchronous motor using PID and LQR," in *Proc. 8th Int. Conf. on Computing, Communication and Networking Technologies (ICCCNT)*, 2017, pp. 1-6.

[39] L. T. Seng, T. H. Lee, and K. T. Chang, "An optimal speed controller for permanent-magnet synchronous motor drives," *IEEE Trans. Ind. Electron.*, vol. 41, no. 5, pp. 503-510, 1992.
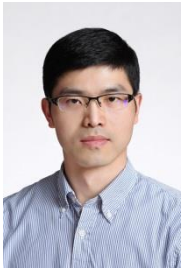
Xiaodong Sun (M'12-SM'18) received the B.Sc. degree in electrical engineering, and the M.Sc. and Ph.D. degrees in control engineering from Jiangsu University, Zhenjiang, China, in 2004, 2008, and 2011, respectively.

Since 2004, he has been with Jiangsu University, where he is currently a Professor with the Automotive Engineering Research Institute. From 2014 to 2015, he was a Visiting Professor with the School of Electrical, Mechanical, and Mechatronic Systems, University of Technology Sydney, Sydney, Australia. His current teaching and research interests include electrical machines and drives, drives and control for electric vehicles, and intelligent control. He is the author or coauthor of more than 80 refereed technical papers and one book, and he is the holder of 36 patents in his areas of interest.

Changchang Hu was born in Xuzhou, Jiangsu, China, in 1994. He received the B.S. degree in vehicle engineering from Jinling Institute of Technology, Nanjing, China, in 2017, and he is currently working toward the M.E. degree in Jiangsu University, Zhenjiang, China.

His current research interests include control of electrical drive systems and intelligent algorithm for automobile application.

Gang Lei (M'14) received the B.S. degree in Mathematics from Huanggang Normal University, China, in 2003, the M.S. degree in Mathematics and Ph.D. degree in Electrical Engineering from Huazhong University of Science and Technology, China, in 2006 and 2009, respectively. He is currently a senior lecturer at the School of Electrical and Data Engineering, University of Technology Sydney (UTS), Australia. His research interests include design optimization and control of electrical drive systems and renewable energy systems.

Youguang Guo (S'02-M'05-SM'06) received the B.E. degree from Huazhong University of Science and Technology, China in 1985, the M.E. degree from Zhejiang University, China in 1988, and the Ph.D. degree from University of Technology, Sydney (UTS), Australia in 2004, all in electrical engineering. He is currently a professor at the School of Electrical and Data Engineering, University of Technology Sydney (UTS). His research fields include measurement and modeling of properties of magnetic materials, numerical analysis of electromagnetic field, electrical machine design optimization, power electronic drives and control.

Jianguo Zhu (S'93–M'96–SM'03) received the B.E. degree in 1982 from Jiangsu Institute of Technology, Jiangsu, China, the M.E. degree in 1987 from Shanghai University of Technology, Shanghai, China, and the Ph.D. degree in 1995 from the University of Technology Sydney (UTS), Sydney, Australia, all in electrical engineering. He was appointed a lecturer at UTS in 1994 and promoted to full professor in 2004 and Distinguished Professor of Electrical Engineering in 2017. At UTS, he has held various leadership positions, including the Head of School for School of Electrical, Mechanical and Mechatronic Systems and Director for Centre of Electrical Machines and Power Electronics. In 2018, he joined the University of Sydney, Australia, as a full professor and Head of School for School of Electrical and Information Engineering. His research interests include computational electromagnetics, measurement and modelling of magnetic properties of materials, electrical machines and drives, power electronics, renewable energy systems and smart micro grids.