Int J Adv Manuf Technol (2016) 86:3123–3134
DOI 10.1007/s00170-016-8406-x

ORIGINAL ARTICLE

Predictive DTC schemes with PI regulator and particle swarm optimization for PMSM drive: comparative simulation and experimental study

H. Mesloub[1] · M. T. Benchouia[1] · A. Goléa[1] · N. Goléa[2] · M. E. H. Benbouzid[3]

Received: 11 October 2015 / Accepted: 19 January 2016 / Published online: 10 February 2016

Abstract This paper shows that a predictive control (PC) combined with the principle of direct torque control (DTC) leads to an excellent dynamic behavior of the permanent magnets synchronous machine (PMSM) and mitigates the drawbacks of the conventional DTC. In this paper, a predictive direct torque control (PDTC) of a PMSM based on classical PI regulator and particle swarm optimization (PSO) algorithm is developed. The PSO adjusts the parameters of the PI controller, which improves its adjustive capability. The effectiveness of the proposed PDTC based a PSO controller is then compared with that of the conventional PI controller. The numerical simulation is performed using MATLAB-Simulink and SimPowerSystem Toolbox. The main contribution of this work is the implementation of the proposed controller on a test bench around a dSPACE 1104. Simulation and experimental results are presented in order to demonstrate the effectiveness of the proposed techniques. Besides, the system associated with these techniques reduces effectively the flux and the torque ripples with better dynamic and steady state performance. The results with the PSO show more performances and a considerable reduction in torque and flux ripples than the results with the conventional PI.

✉ H. Mesloub
sassamesloub@yahoo.fr

[1] LGEB Laboratory, University of Biskra, BP.145, 07000 Biskra, Algeria

[2] LGEA Laboratory, Larbi Ben M'hidi University, 04000 Oum El-Bouaghi, Algeria

[3] LBMS, University of Brest, Brest Cedex 03, France

Keywords Permanent magnet synchronous motor (PMSM) · Direct torque control (DTC) · Particle swarm optimized (PSO) · Predictive control (PC) · Practical validation

1 Introduction

Compared to DC motors, PMSM are more reliable due to the absence of commutators. The PMSM also has many advantages over AC induction motors. PMSM generates the rotor magnetic flux with rotor magnets, so achieves higher efficiency. The classification of the PMSM, their merits and demerits, magnetic characteristics of magnets used and their comparison with induction motors are presented in [1]. Therefore, PMSM are used in electric and hybrid vehicles, high-end white goods (refrigerators, washing machines, dishwashers, etc.), high-end pumps, fans and in other appliances that require high reliability and efficiency [2]. Despite its advantages, such as high efficiency, high power density and high torque to current ratio, PMSM remains complicated and difficult to control when—good transient performance under all operating conditions is desired [3]. This is due to the fact that the PMSM is a nonlinear, multivariable and time varying system subjected to unknown disturbances and variable parameters.

Over the past decades, various control techniques have been developed in order to improve the performance of the PMSM in the presence of external disturbances and parameters variation. However, the two most widespread control schemes employ linear vector control (VC) and the DTC [4–6]. DTC has a relatively simple control structure. It is also known that DTC drive is less sensitive to parameters de-tuning (only stator resistance is used to estimate the stator flux) and provides a high dynamic performance compared to classical VC

(fastest response of torque and flux). The DTC is essentially based on a localization table which allows selecting the inverter switching state according to the position of the stator flux vector and of the direct control of stator flux and electromagnetic torque. DTC allows a decoupled control of flux and torque without using speed or position sensors. Unfortunately, due to hysteresis type controller, DTC produces a variable switching frequency and consequently large torque and flux ripples and high currents distortion, which is more evident at the low speed [7].

Recently, the Predictive Control (PC) was investigated in the PMSM drive system. The PC is an optimization-based approach that computes the next control action by minimizing the difference between the predicted controlled variables and the specified reference [8]. The PC has several merits, such as the easy inclusion of nonlinearities and constraints [9, 10]. The PC is employed to generate the required control signal to implement the DTC technique, while minimizing the torque and the flux ripples and the switching frequency [11]. Constrained the PC of PMSM are studied in [12].

Another field of investigation, to improve the performances of the conventional DTC, was the use of several modifications that have been proposed, such as modified switching tables, SVM modulation, predictive control (PC), particle swarm optimization (PSO), and intelligent control techniques e.g., fuzzy systems logic and neural networks [13, 14]. This is mainly due to their learning ability and generalization capacities. The basic idea behind using intelligent techniques resides in replacing the hysteresis comparators and the switching table with a predictive control (PC), intelligent controller. This results in more smooth transition in control actions and significant flux and torque ripples reduction.

In this paper, a PDTC of the PMSM based on classical PI regulator and PSO algorithm are developed and a comparison study is presented. To show the performances, a numerical simulation is performed. To validate the simulation results, these algorithms are implemented on a test bench around a DSPACE 1104. Comparing these techniques, we see that PDTC with PSO algorithm is more efficient than PDTC with PI regulator.

The remainder of this paper is organized as follows: Section 2 reviews the PMSM modeling and the conventional DTC design principals. Section 3 presents the PDTC approach associated with PI regulator. Section 4 presents the PDTC with PSO approach. Section 5 develops the two techniques evaluation based on extensive simulation and experimental results. Concluding remarks are presented in section 6.

2 DTC principal

The electrical and the mechanical equations of the PMSM in the stator reference frame (α, β) are as follows:

$$\begin{cases} V_{s\alpha} = R_s I_{s\alpha} + L_s \dfrac{d}{dt} I_{s\alpha} - \omega \phi_f \sin(\theta_s) \\ V_{s\beta} = R_s I_{s\beta} + L_s \dfrac{d}{dt} I_{s\beta} + \omega \phi_f \cos(\theta_s) \end{cases} \quad (1)$$

$$\begin{cases} \dfrac{d}{dt} \phi_{s\alpha} = V_{s\alpha} - R_s I_{s\alpha} \\ \dfrac{d}{dt} \phi_{s\beta} = V_{s\beta} - R_s I_{s\beta} \end{cases} \quad (2)$$

And the electromagnetic torque T_e is given by:

$$T_e = \frac{3}{2} P \left[\phi_{s\alpha} I_{s\beta} - \phi_{s\beta} I_{s\alpha} \right] \quad (3)$$

The equation for the motor dynamics, on the other hand, is:

$$T_e - T_r - f\Omega = J \frac{d\Omega}{dt} \quad (4)$$

The main idea of the DTC is to directly control the torque and flux produced by the machine, without current control, as it is the case in Field Oriented Control (FOC) [15, 16]. Different DTC approaches have been developed [17]. Figure 1 is a typical DTC system. Usually a DC bus voltage

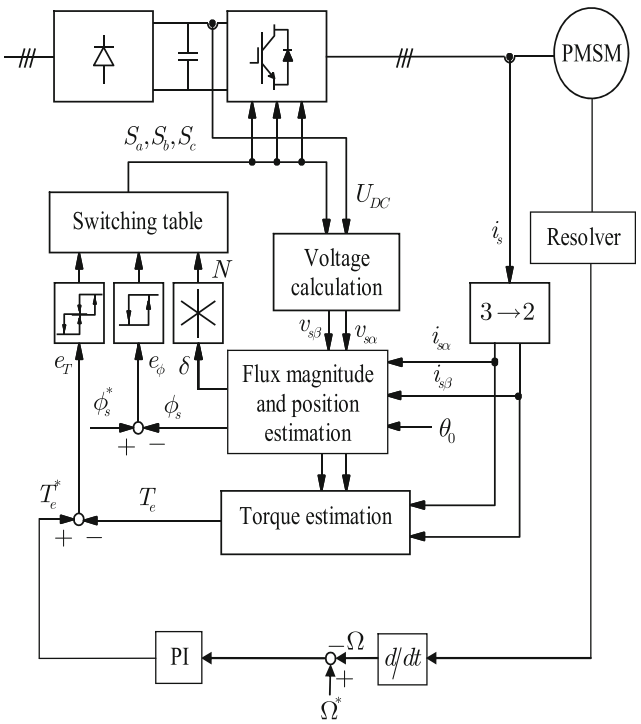


Fig. 1 PMSM DTC block diagram

sensor and two-phase current sensors are needed for the flux and torque estimator. The stator flux amplitude $\phi_s = \sqrt{\phi_{s\alpha}^2 + \phi_{s\beta}^2}$ and position $\delta = \tan^{-1}(\phi_{s\beta}/\phi_{s\alpha})$ are computed from the flux components estimation given by:

$$\begin{cases} \phi_{s\alpha} = \int_0^{T_s} (V_{s\alpha} - R_s I_{s\alpha})dt + \phi_{s\alpha_0} \\ \phi_{s\beta} = \int_0^{T_s} (V_{s\beta} - R_s I_{s\beta})dt + \phi_{s\beta_0} \end{cases} \quad (5)$$

Where $\phi_{s\alpha_0}$ and $\phi_{s\beta_0}$ are the initial stator flux values.

The Torque is estimated from (3).

The switching state of the inverter is updated, at each sampling time T_s, depending on flux and torque hysteresis comparators outputs and stator flux position sector as shown in Fig. 2 and Table 1. Therefore, the switching frequency is usually not fixed; it changes with the rotor speed, load and bandwidth of the flux and torque controllers. The main advantages of DTC are absence of coordinate transformation and current regulator; absence of separate voltage modulation block. Common disadvantages of conventional DTC are high torque ripple and slow transient response to the step changes in torque during startup.

3 PDTC approach based on PI regulator

In this paper, we will design the DTC associated with the so-called PC [18]. To do this, we assume that the process is mono-variable and there are no constraints to be respected. Figure 3 illustrates the basic idea of the

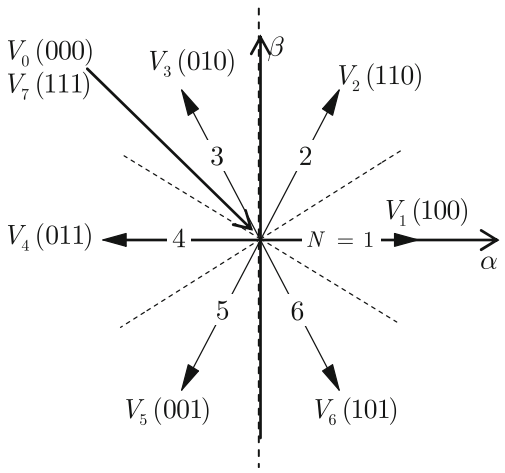


Fig. 2 Two-level VSI voltage vectors and sectors

Table 1 PMSM DTC switching table

		Sector N					
e_ϕ	e_T	1	2	3	4	5	6
1	1	V_2	V_3	V_4	V_5	V_1	V_4
	0	V_7	V_0	V_7	V_0	V_7	V_0
	−1	V_6	V_1	V_2	V_3	V_4	V_5
0	1	V_3	V_4	V_5	V_6	V_1	V_2
	0	V_7	V_0	V_7	V_0	V_7	V_0
	−1	V_6	V_1	V_2	V_3	V_4	V_5

PC. The PC is based on a priori knowledge of the process through a model that provides predictions of changes in future outings. This prediction is then compared to the desired output of a finite horizon, called the prediction horizon *Np*. The computer then determines the optimal sequence of controls to minimize the difference between the predicted output and the reference, but only the first component is actually implemented. At the next sampling instant, the prediction horizon and not a slip of the optimization problem are repeated and so on. Therefore, this control strategy is called receding horizon control.

The basic idea of the PC is to predict the future behavior of the variables over a time frame based on the model of the system. In fact, PC is an extension of DTC, as it replaces the look-up table with an online optimization process in the control of machine torque and flux. It is different from the employments of hysteresis comparators and switching table in conventional

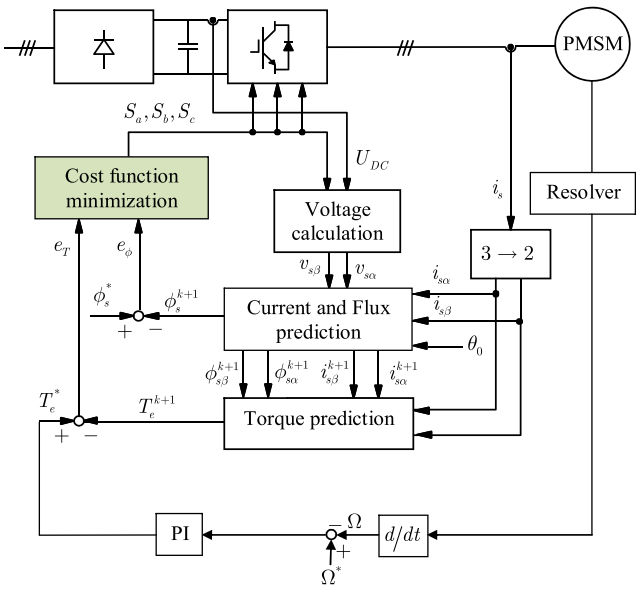


Fig. 3 PDTC scheme with PI regulator

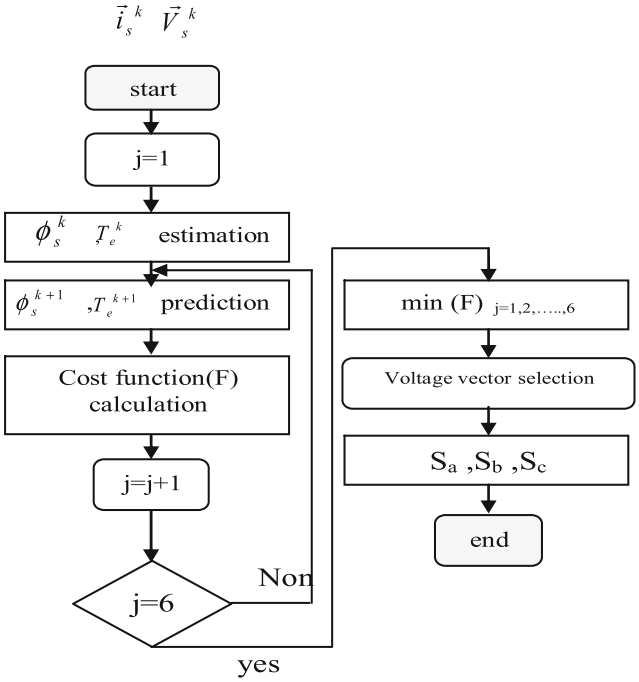


Fig. 4 Flowchart of the improved PDTC

DTC. The principle of vector selection in PC is based on evaluating a defined cost function. The selected voltage vector from conventional switching table is not necessarily the best one in terms of reducing torque and flux ripples. There are limited discrete voltage vectors in the two-level inverter-fed PMSM drives, as a result, it is possible to evaluate the effects of each voltage vector and select the one minimizing the cost function.

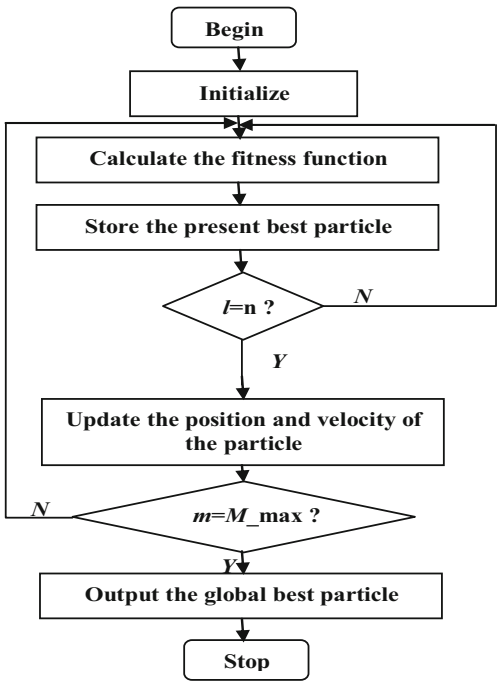


Fig. 5 Program flow chart

Table 2 Result of the fitness values after generation

Literation	Parameter	Fitness
1	Velocity (1,1)⟹Kp = current_position(1,1) = 0.698 Velocity (2,1)⟹Ki = current_position(2,1) = 0.701	0.1400
7	Velocity (1,1)⟹Kp = current_ position(1,1) = 0.6000 Velocity (2,1)⟹Ki = current_ position(2,1) = 0.6500	0.040
16	Velocity (1,1)⟹Kp = current_ position(1,1) = 0.0300 Velocity (2,1)⟹Ki = current_ position(2,1) = 0.5000	0.018
20	Velocity (1,1)⟹Kp = current_ position(1,1) = 0.0100 Velocity (2,1)⟹Ki = current_ position(2,1) = 0.6000	0.018

As shown in Fig. 3, PC includes three parts: cost function minimization, predictive model and flux and torque estimators.

3.1 Cost function minimization

For the two-level inverter-fed PMSM drives, the prediction means the effect of each voltage vector when applied to the machine. For PC, the cost function is such chosen that both torque and flux at the end of the cycle is as close as possible to the reference value. Generally, the minimum value of cost function is defined as.

$$F = \left|T_e^* - T_e^{k+1}\right| + k_1 \left|\left|\phi_s^*\right| - \left|\phi_s^{k+1}\right|\right| \quad s.t. \quad u_s^k \in \{V_1, V_2, \cdots V_5, V_6\} \quad j = 1 \cdots 6 \tag{6}$$

Where T_e^* and ϕ_s^* are the reference values of torque and flux respectively, T_e^{k+1} and ϕ_s^{k+1} are predicted values of torque and flux, respectively, k_1 is the

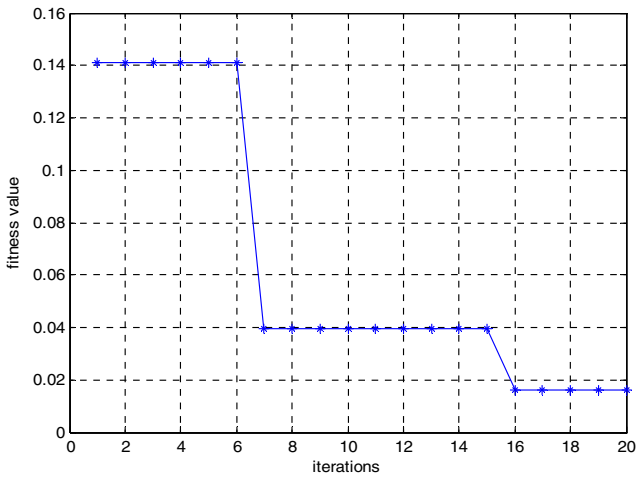


Fig. 6 Optimization process of the PSO

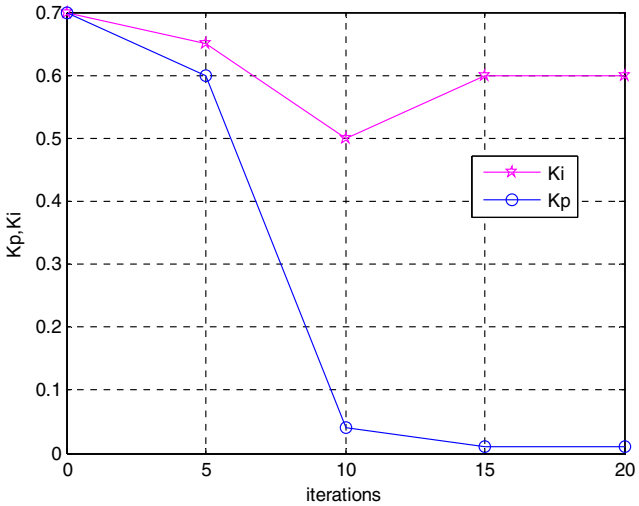


Fig. 7 Adjust the PI parameter

weighting factor. *V1*, *V2*, *V3*, *V4*, *V5*, and *V6* are six nonzero voltage space vectors and can be generated by three phase inverter with respect to the different switches states. A set of voltage space vectors is defined as :

$$u_s^k = \sqrt{\frac{2}{3}} U_{dc}\left[S_a^k + S_b^k \cdot e^{i\frac{2\pi}{3}} + S_c^k \cdot e^{i\frac{4\pi}{3}}\right] \tag{7}$$

Where $S_x^k (x = a, b, c)$ is upper power switch state of one of three legs.

$S_x^k = 1$ *or* $S_x^k = 0$: when the upper power switch of one leg is on or off.

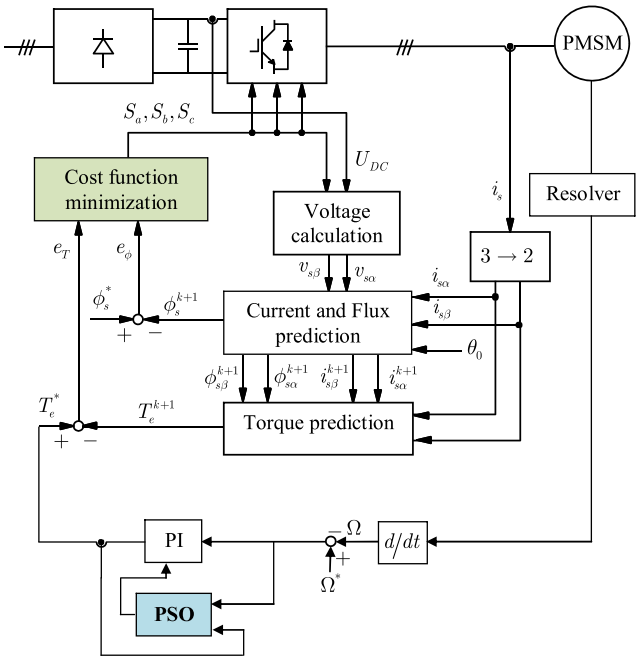


Fig. 8 PDTC scheme with PSO_PI regulator for PMSM

Table 3 Main parameters of the prototype

PMSM parameters		
Number of pairs of poles	p	2
Stator winding resistance	R_s	2.6 Ω
Nominal Current	I_N	4.5 A
Inductance d-axis Ld	L_d	43 mH
Inductance d-axis Lq	L_q	43 mH
PM flux linkage	ϕ_f	0.178 Wb
Moment of inertia	J	$85e^{-6}$ kg/m^2

3.1.1 Predictive model for stator currents

According to (1), the prediction of the stator current at the net sampling instant is expressed as

$$\begin{cases} I_{s\alpha}^{k+1} = I_{s\alpha}^k + \dfrac{1}{L_s}\left[-R_s I_{s\alpha}^k + \hat{\omega}^k \phi_f \sin\left(\theta_s^k\right) + V_{s\alpha}^k\right] \cdot T_s \\ I_{s\beta}^{k+1} = I_{s\beta}^k + \dfrac{1}{L_s}\left[-R_s I_{s\beta}^k - \hat{\omega}^k \phi_f \cos\left(\theta_s^k\right) + V_{s\beta}^k\right] \cdot T_s \end{cases} \tag{8}$$

Where T_s is the sampling period.

3.1.2 Torque and flux estimators

The stator flux linkage ϕ_s , in the reference mark (α-β) is given by:

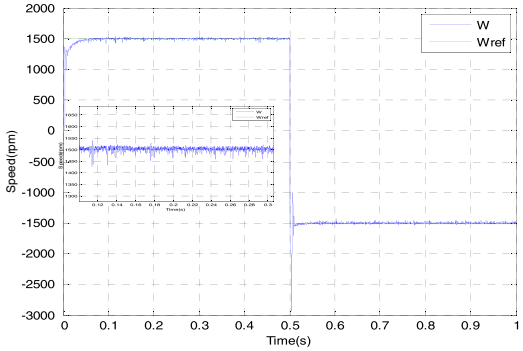
$$\begin{cases} \phi_{s\alpha_{est}}^{k+1} = \phi_{s\alpha}^k + \left(V_{s\alpha}^k - R_s I_{s\alpha}^{k+1}\right) \cdot T_s \\ \phi_{s\beta_{est}}^{k+1} = \phi_{s\beta}^k + \left(V_{s\beta}^k - R_s I_{s\beta}^{k+1}\right) \cdot T_s \\ \phi_s^{k+1} = \sqrt{\left(\phi_{s\alpha_{est}}^{k+1}\right)^2 + \left(\phi_{s\beta_{est}}^{k+1}\right)^2} \end{cases} \tag{9}$$

The electromagnetic torque developed in (α-β) system can be estimated as following:

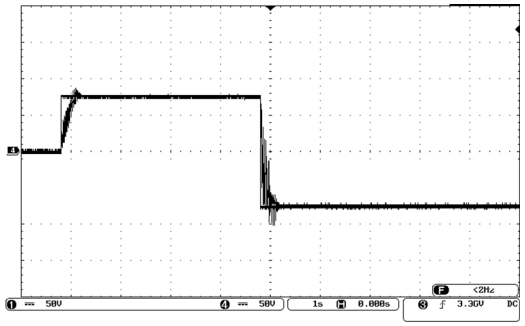
$$T_e^{k+1} = \frac{3}{2} P\left[\phi_{s\alpha_{\text{est}}}^{k+1} I_{s\beta}^{k+1} - \phi_{s\beta_{\text{est}}}^{k+1} I_{s\alpha}^{k+1}\right] \tag{10}$$

The basic operation of the predictive controller is summarized by the following steps.

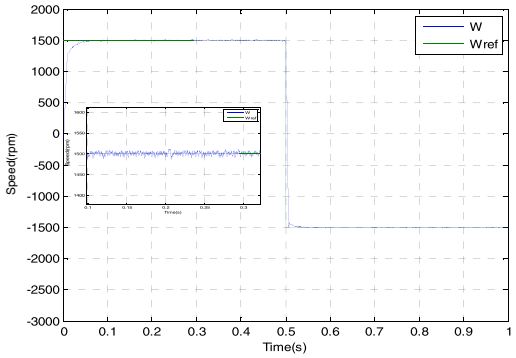
1. Stator currents and the stator voltages are measured.
2. These measurements are used for prediction of torque and stator flux for all seven different voltage vectors.
3. The seven predictions are evaluated using the cost function (cost function block).



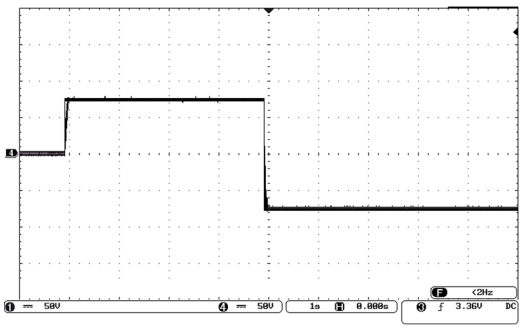
Simulation results *PDTC_PI*: Mechanical speed (rpm).



Experimental results *PDTC_PI:* Mechanical speed, scale: Ch1, Ch4 (1000 rpm / div). Time scale: 1s/div.



Simulation results *PDTC_PSO*: Mechanical speed (rpm).



Experimental results *PDTC_PSO*: Mechanical speed, scale: Ch1, Ch4 (1000 rpm / div). Time scale: 1s/div.

Fig. 9 Comparison graph of speed (rpm)

4. The voltage vector that minimizes the cost function is selected and applied in the machine terminals.

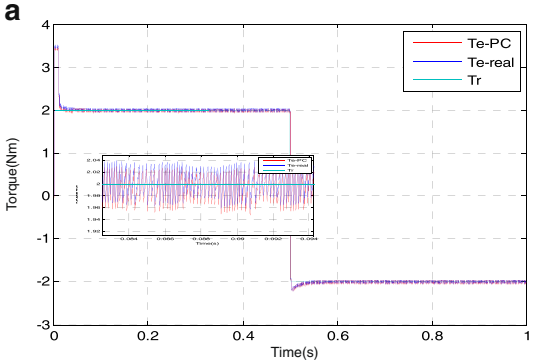
These steps are repeated each sampling time, taking into account new measurements and references. Closed loop control is obtained through the feedback of measurements used for prediction and the action decision taken to minimize the value of the cost function F (Fig. 4).

4 PDTC approach with PSO-PI algorithm

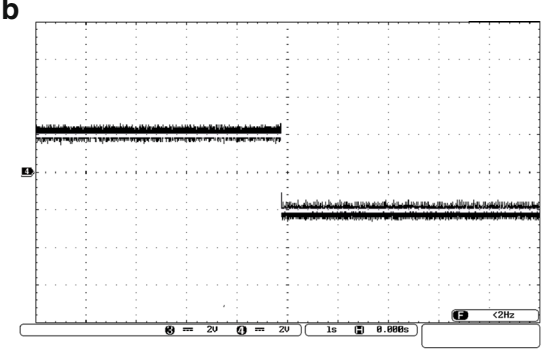
The PSO technique can generate a high-quality solution within shorter calculation time and stable convergence characteristic than other stochastic methods [19]. Because the PSO method is an excellent optimization methodology and a promising approach for solving the optimal PI controller parameters problem [20–22]; therefore, this study develops the PSO-PI controller to search optimal PI parameters. This PI controller is called the PSO-PI controller.

The particles evaluate their positions relative to a goal (fitness) at each iteration. The particles in a local neighborhood share memories of their "best" positions, and then use those memories to adjust their own velocities, and the subsequent positions. Each particle keeps track of its coordinates in the problem space, which are associated with the best solution (evaluating value) it has achieved so far. This value is called pbest. Another best value that is tracked by the global version of the particle swarm optimizer is the overall best value, and its location, obtained so far by any particle in the group, is called gbest.

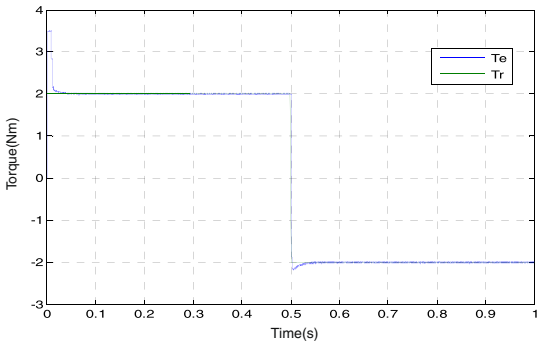
The trajectory of each particle in search domain is adjusted by dynamically altering the velocity of each particle, according to its own flying experience and the flying experience of the other particles. In each iteration, the next position of each particle is determined



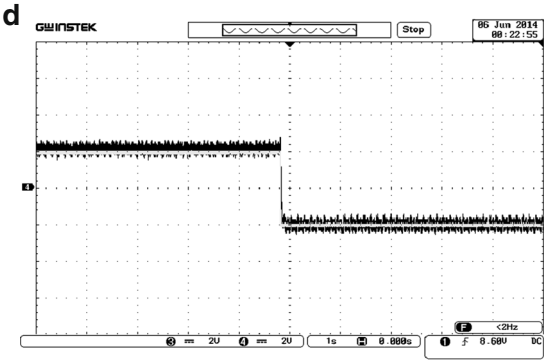
Simulation results *PDTC_PI*: Torque response (Nm).



Experimental results *PDTC_PI*: Torque response scale: Ch3, Ch4 (2Nm/div). Time scale: 1s/div.



Simulation results *PDTC_PSO*: Torque response (Nm).



Experimental results *PDTC_PSO*: Torque response scale: Ch3,Ch4 (2Nm/div). Time scale: 1s/div.

Fig. 10 Comparison graph of torque (Nm)

by its current position $x_p(t)=(x_{p1}(t),\ \ldots,\ x_{pn}(t))^T$ and velocity $v_p(t+1)=(v_{p1}(t+1),\ldots,\ v_{pn}(t+1))^T$ as follows:

$$x_p(t+1) = x_p(t) + v_p(t+1) \tag{11}$$

where t denotes the iteration time. The next velocity of each particle is determined by its current velocity $v_p(t)$ (momentum), the best position ever found by the *p-th* particle $\text{pbest}_p(t)$ (memory), and best position ever found by the swarm $\text{gbest}(t)$ (shared information) as follows:

$$\begin{aligned} v_{pi}(t+1) = {} & \omega v_{pi}(t) + C_1 \text{rand}_1 \left(\text{pbest}_{pi}(t) - x_{pi}(t)\right) \\ & + C_2 \text{rand}_2 \left(\text{gbest}_i(t) - x_{pi}(t)\right) \end{aligned} \tag{12}$$

At each time step, each particle chooses the best of the best performances in its possession, changes its velocity based on this information and its own data and moves accordingly.

The particles may progressively converge towards an optimum: the ability to converge near the best optimal solution and have the best convergence time.

4.1 PSO-PI implementation

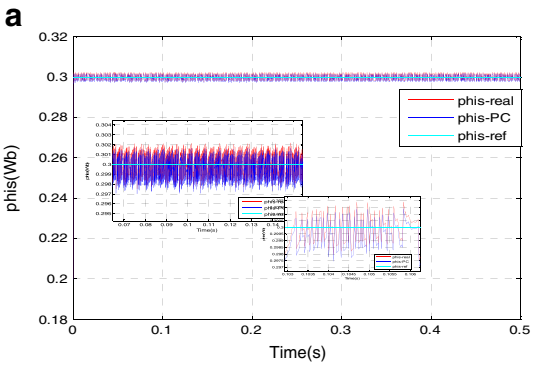
The basic principle of PSO controller used can be described as follows. Considering a group which consists of n particles, each particle searches the best position under a certain velocity. It updates its position according to the best record of its own and others in the history.

The current position of particle "*l*" is represented as

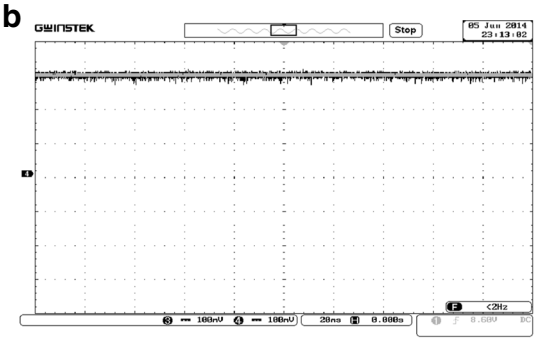
$$K^l = \left(K_p^l,\ K_i^l\right) \quad l = 1,\ 2,\ \ldots,\ n \tag{13}$$

The current velocity of particle "*l*" is represented as

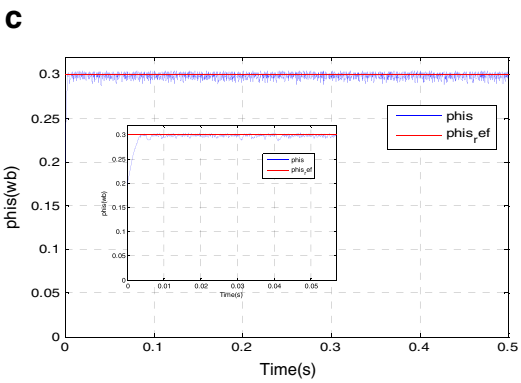
$$V^l = \left(V_p^l,\ V_i^l\right) \quad l = 1,\ 2,\ \ldots,\ n \tag{14}$$



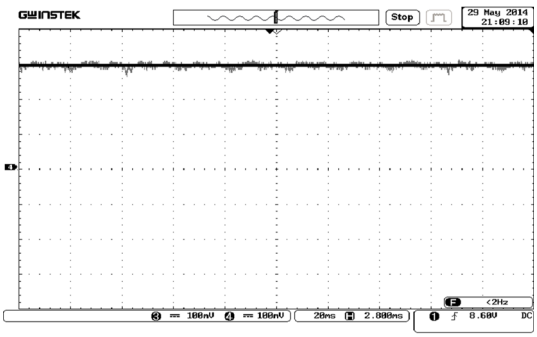
Simulation results *PDTC_PI*: stator Flux response (Wb)



Experimental results *PDTC_PI*: stator Flux response scale: Ch2, Ch4 (0.1Wb/div) Time scale: 20ms/div .



Simulation results *PDTC_PSO*: Stator Flux response (Wb).



Experimental results *PDTC_PSO*: stator Flux response scale: Ch3, Ch4 (0.1Wb/div). Time scale: 20ms/div.

Fig. 11 Comparison graph of stator flux (Wb)

The best position of particle "*l*" in its search history is represented as

$$P^l = \left(P_p^l,\ P_i^l\right) \quad l = 1,\ 2,\ \ldots,\ n \tag{15}$$

The best position of the population in its search history is represented as

$$P_{best}^g = \left(P_p^g,\ P_i^g\right) \quad l = 1,\ 2,\ \ldots,\ n \tag{16}$$

Update the velocity and position by the following equations

$$\begin{aligned} V^{l+1} &= w{\cdot}V^l + C_1{\cdot}R_1{\cdot}\left[P^l{-}K^l\right] + C_2{\cdot}R_2{\cdot}\left[P_{\text{best}}^g{-}K^l\right] \\ K^{l+1} &= K^l + V^{l+1} \quad l = 1,\ 2,\ \ldots,\ n \end{aligned} \tag{17}$$

Where ω is the constant inertia weight, C_1, C_2 is the learning factor (C_1, C_2 is usually on [0, 4] interval); R_1, R_2 are uniformly distribution pseudo-random numbers on [0, 1] interval. The velocity of a particle is usually limited to a maximum speed. It can prevent the system unstable from the affect of some bad particles. In General, the motor needs the PI parameters be tuned at the appropriate value quickly during startup. Then make fine adjustments according to the load torque to improve the performance of the system. So we use time-varying weights in the speed updating (16) to substitute the constant weight and set the weight range on $[\omega_{\max}, \omega_{\min}]$ interval. At each sampling time, the population iterates $M\_\max$ times. The iteration in the time "m" of the inertia weight is

$$\omega^m = \omega_{\max} - \frac{\omega_{\max} - \omega_{\min}}{M\_\max}{\cdot}m \tag{18}$$

Ring topology is used as the neighbor topology of particle swarm. The influence of neighbors is delivered one by one until the best particle is found. The fitness function composed

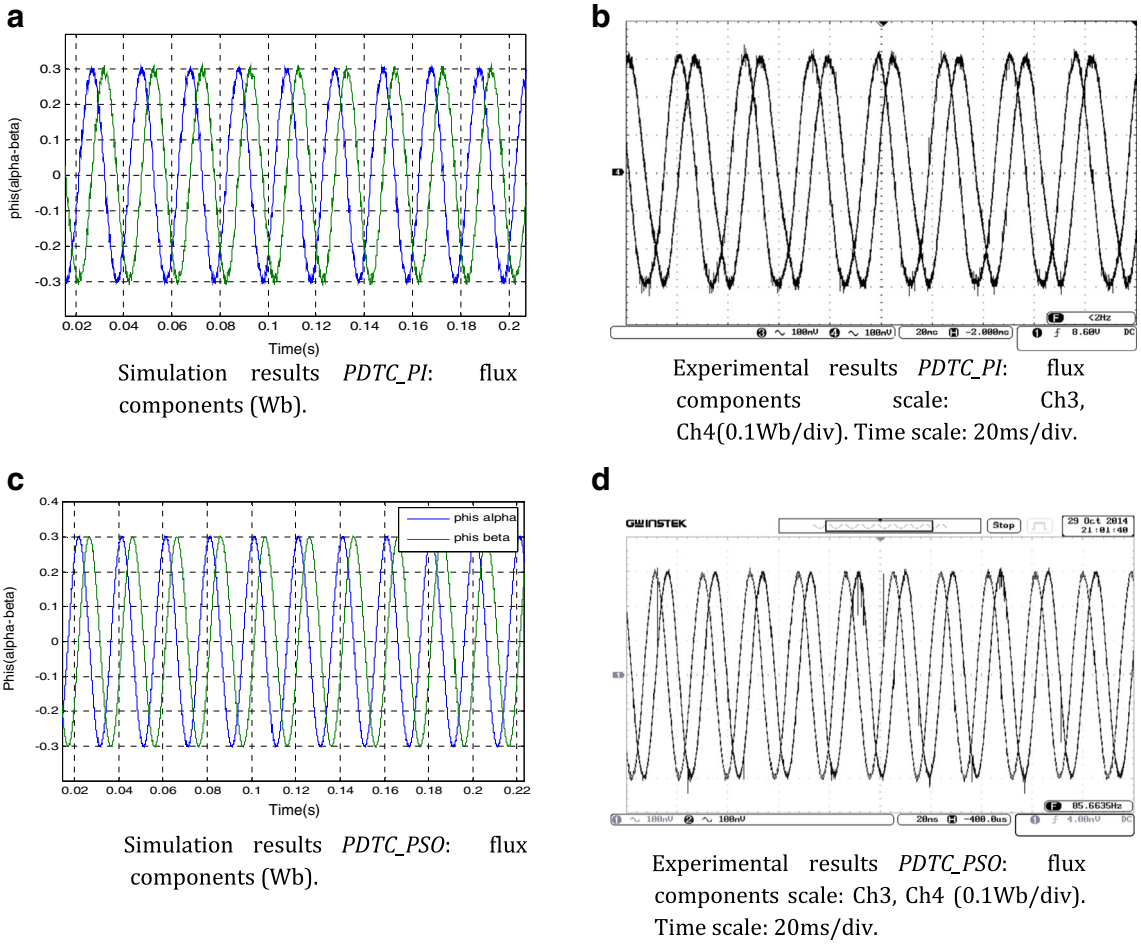


Fig. 12 Comparison graph of stator flux components (Wb)

of the speed error (e_w) and the speed error change (Δe_w) of PMSM is as

$$F = R_1 \cdot |e_w(l)| + R_2 \cdot |\Delta e_w(l)| \quad R_1, R_2 \in [0, 1] \tag{19}$$

The position value (K_p, K_i) of this particle in the search space is the optimal PI parameters. The pseudo-code of PSO tuning algorithm is given below (Fig. 5 and Table 2).

From Fig. 6, it can be seen that the algorithm usually converges within 20 iterations because of a good initial guess. But from this figure we can also see that the initial fitness is not as good as others. This is because the fitness's sensitivity to parameter changes is different at different operating points (Fig. 7). Sometimes, even a small change in the parameters will result in a large change in the fitness.

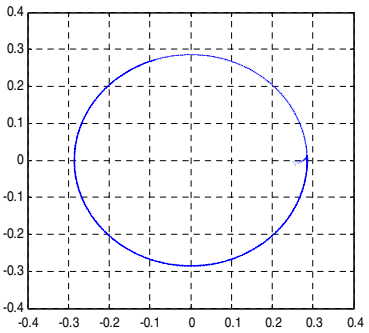
To improve the performances of PMSM, a PDTC method based on a PSO is used. The PSO concept adjusts the real–time parameters of Kp and Ki. In addition, the DTC_PC with PSO controller increased dynamic response due to the real-time of the control. The leading PSO algorithm also shows its fast convergence rate and the ability of avoiding the local optimal points.

The basic structure of the proposed algorithm is shown in Fig. 8. The main blocks will be illustrated in the following.

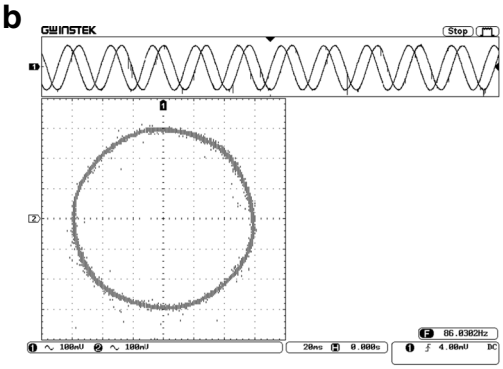
5 Simulation and experimental results

A simulation model of the system was established by Simulink in MATLAB. A comparative study using simulation results of the PDTC based on a conventional PI regulator and PSO-PI was carried out. The speed and flux references used in the simulation are (+1500 rpm in 0 s, −1500 rpm in 0.5 s) and 0.3 Wb, respectively. The experimental device based on DSP 1104 was developed and the experimental results are

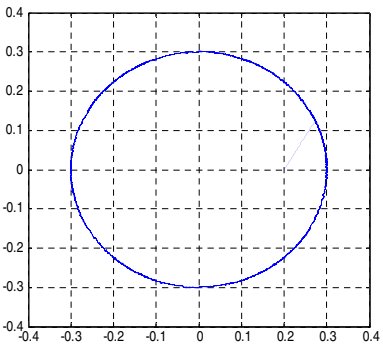
Fig. 13 Comparison graph of flux trajectory ($\Phi_{s\alpha}$, $\Phi_{s\beta}$) (Wb)



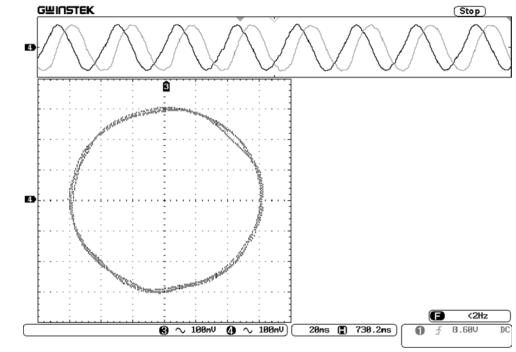
Simulation results *PDTC_PI*: Trajectory of flux (Φs_α, Φs_β) (Wb).



Experimental results *PDTC_PI*: Trajectory of flux (Φs_α, Φs_β) scale: Ch3, Ch4 (0.1 Wb/div). Time scale: 20ms/div.



Simulation results *PDTC_PSO*: Trajectory of flux (Φs_α, Φs_β) (Wb).



Experimental results *PDTC_PSO*: Trajectory of flux (Φs_α, Φs_β) scale: Ch3,Ch4(0.1Wb/div).Timescale: 20ms/div.

compared to the simulation results. The sampling period of the system is 10^{-4} ms (Table 3).

To compare the performances of the PDTC_PI with the performances of the PDTC_PSO-PI a simulation and experimental results are presented. In two cases, the dynamic responses of Speed, flux and torque for the starting process with 2 Nm load torque applied and a constant command flux of 0.3 Wb are shown in Figs. 9, 10, and 11.

Figure 9a, b show the simulation and the experimental speed response in the PDTC_PI case. Figure 9c, d d show the simulation and the experimental speed response in the PDTC_PSO-PI case. The speed of the PDTC with PSO algorithm and system response converges more quickly than the PDTC with PI control system.

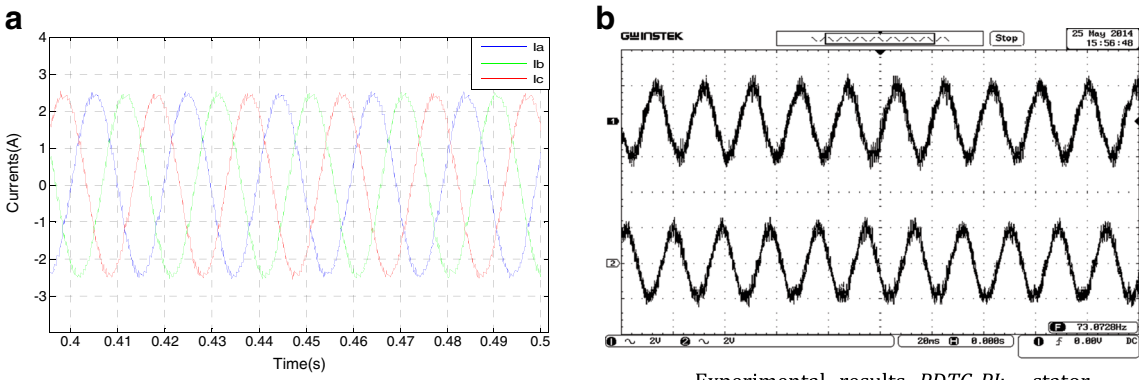
The torque responses are presented in Fig. 10a, b for PDTC_PI and in Fig. 10c, d for PDTC_PSO-PI, respectively. The results show a better dynamic and steady state performances in the PDTC_PSO-PI case compared to the PDTC_PI. The torque ripples are significantly reduced at the same operating conditions.

Figure 11a–d shows the response of the stator flux magnitude for both methods PDTC_PI and PDTC_PSO. Figure 11c, d, the stator flux, is the fast response in transient state and the ripple in steady state is reduced remarkably compared with PDTC_PI.

The flux changes through big oscillation and the torque ripple is bigger in PDTC_PI. Figure 11a, b shown it can be seen that the torque and flux ripple of PMSM is reduced greatly in the DTC Predictive with PSO. It can be seen from simulation and experimental results.

Figure 12a–d, shows the simulation and the experimental results of the flux components for both methods respectively.

From Fig. 13a–d, it can be noticed that the stator_flux vector describes a circular trajectory for both cases.



Simulation results *PDTC_PI*: stator currents (A) at 1500 rpm.

Experimental results *PDTC_PI*: stator currents at 1500 rpm scale: Ch1, Ch2 (2A/div). Time scale: 20ms/div.

Simulation results *PDTC_PSO*: stator currents (A) at 1500 rpm.

Experimental results *PDTC_PSO:* stator currents at 1500 rpm scale: Ch1,Ch4 (2A/div). Time scale: 20ms/div.

Fig. 14 Comparison graph of stator currents (A)

The flux components response shows that the flux ripples are greatly reduced in the PDTC_PSO-PI compared to the PDTC_PI.

Finally, as a consequence of the reduction of the torque and the flux ripples, the waveform of the stator current is improved. Figure 14a–d shows the steady state current response for the both methods.

Consequently, the stator current With the PDTC_PSO-PI is very smooth compared to the PDTC_PI case, so the stator current is nearly sinusoidal waveform, thereby minimizing the harmonics.

6 Conclusion

This paper describes the design and the implementation of a direct torque controlled PMSM, based on the PC with classical PI regulator and the PC with PSO algorithm. These advanced techniques are used in order to improve the performances of the DTC, in particular reduction of torque and flux ripples, high dynamic response and the switching frequency reduction. To show the performances of the proposed methods, simulations were performed using Matlab-Simulink. To validate the simulation results, a test bench has been constructed based on the DSP-1104. The simulation and the experimental results show a good performances for both methods compared to classical DTC. The PDTC with PSO algorithm shows more performances than the PDTC with classical PI regulator, in particular the reduction of the torque and the flux ripples.

Simulation and experimental results both show that compared with DTC_PC with PI regulator, the flux and torque ripples are reduced greatly in PMSM DTC based on predictive control with particle swarm optimization algorithm, and the switching frequency of the inverter is constant, improving the control performance of the system.

References

1. Tulasi Ram Das G (2008) JNTUCE, Hyderabad, Introduction and control of permanent magnet synchronous machines, Workshop on 'Modern AC Drives' at RGMCET, Nandyala
2. Finch JW, Giaouris D (2008) Controlled AC electrical drives. IEEE Trans Ind Electron 55(2):481–491
3. Rodriguez J, Kennel RM, Espinoza JR, Trincado M, Silva CA, Rojas CA (2012) High-performance control strategies for electrical drives: an experimental assessment. IEEE Trans Ind Electron 59(2): 812–820
4. Zhang Y, Akujuobi CM, Ali WH, Tolliver CL, Shieh L-S (2006) Load disturbance resistance speed controller design for PMSM. IEEE Trans Ind Electron 53(4):1198–1208
5. Zhang Y, Shieh L-S, Akujuobi C, Gu X (2005) Observer-based load disturbance compensation for motor drive with DSP implementation. IEEE-IES, 31st Annual Meeting. pp. 1486 – 1491
6. Dongemei X, Daokui Q, Fang X (2005) Design of H8 feedback controller and IP position controller of PMSM servo system. IEEE International Conference, Mechatronics and Automation 2(29): 578–1108
7. Ossoufi B, Karim M, Ionita S, Lagrioui A (2010) Performance analysis of direct torque control (DTC) for synchronous machine permanent magnet (PMSM). In: Proc. IEEE-SIITME'2010. Pitesti, Romania. pp. 275–280, 23–26
8. Chai S et al (2013) Model predictive control of a permanent magnet synchronous motor with experimental validation. Control Eng Pract 21:1584–1593
9. Errouissi R, Ouhrouche M (2010) Nonlinear predictive controller for a permanent magnet synchronous motor drive. Math Comput Simul 81:394–406
10. Kouro S, Cortes P, Vargas R, Ammann U (2009) Model predictive control: a simple and powerful method to control power converters. IEEE Trans Ind Electron 56:1826–1838
11. Zhu H, Xiao X, Li Y (2012) Torque ripple reduction of the torque predictive control scheme for permanent-magnet synchronous motors. IEEE Trans Ind Electron 59(2):871–877
12. Bolognani S, Peretti L, Zigliotto M (2009) Design and implementation of model predictive control for electrical motor drives. IEEE Trans Ind Electron 56(6):1925–1936
13. Kumar V, Gaur P, Mittal AP (2014) ANN based self tuned PID like adaptive controller design for high performance PMSM position control. Expert Systems with Applications 41:7995–8002
14. Kashif SAR, Saqib MA (2014) Sensorless control of a permanent magnet synchronous motor using artificial neural network based estimator—an application of the four-switch three-phase inverter. Electric Power Components and Systems 42
15. Takahashi I, Noguchi T (1986) A new quick response and high-efficiency control strategy of an induction motor. IEEE Trans Ind Appl 22(5):820–827
16. Boldea (2000) Direct torque and flux control (DTFC) of A.C. drives—a review, in Proceedings of EPEPEMC'2000, Vol. 1, Kosice, Slovakia. pp. 88–97
17. Pacas M, Weber J (2005) Predictive direct torque control for the PM synchronous machine. IEEE Trans Ind Electron 52(5):1350–1356
18. Morel F et al (2008) A predictive current control applied to a permanent magnet synchronous machine, comparison with a classical direct torque control. Electr Power Syst Res 78:1437–1447
19. Ray RN, Chatterjee D, Goswami SK (2009) An application of PSO technique for harmonic elimination in a PWM inverter. Appl Soft Comput 9:1315–1320
20. Kao C-C, Chuang C-W, Fung R-F (2006) The self-tuning PID control in a slider–crank mechanism system by applying particle swarm optimization approach. Mechatronics 16:513–522
21. Liu L, Liu W, Cartes DA (2008) Particle swarm optimization-based parameter identification applied to permanent magnet synchronous motors. Eng Appl Artif Intell 21:1092–1100
22. Wain R-J, Lin Y-F, Chuang K-L (2014) Total sliding-mode-based particle swarm optimization control for linear induction motor. J Franklin Inst 351:2755–2780