International Journal of Current Engineering and Technology
E-ISSN 2277 – 4106, P-ISSN 2347 - 5161

Available at http://inpressco.com/category/ijcet

Research Article

Improved Stator Flux Response of a DTC Induction Motor drive using PSO

Trishna Doley[Ȧ*]

[Ȧ]Deptt. Of Electrical Engineering, Indian School of Mines, Dhanbad

Accepted 01 April 2014, Available online 15 April 2014, **Vol.4, No.2 (April 2014)**

Abstract

Particle Swarm Optimization (PSO) is an excellent stochastic approach based on swarm intelligence which is used for solving continuous and discrete optimization problems. This simple and effective technique can be applied to induction motor drives to get ideal performance. Hence, this paper presents a Direct Torque Controller (DTC) based on PSO. The main objective of this work is to have an improved stator flux response by optimizing the parameters (Kp, Ki) of the Proportional- Integral (PI) controller using PSO technique. A DTC model is constructed with a PSO PI controller and simulated in MATLAB/SIMULINKR. The results of simulation show nearly good performance of the PSO tuned PI controller.

Keywords: *Particle swarm optimization technique, Direct torque controller, PI controller.*

1. Introduction

Induction motors have simple and robust construction, high reliability, low cost and maintenance. So they are extensively used in industries for variable speed application in a wide power range. Hence, a proper controlling of its speed is required to achieve the desired performance for specific application. However, induction motors exhibit non-linearity and time varying characteristics (M.M. Eissa *et al*, 2013) due to which their controlling is not easy to achieve. There are different controlling techniques of induction motor such as scalar control, vector or field oriented control, direct torque control and adaptive control (B.K. Bose, 2011). In this paper the Direct Torque Control (DTC) has been implemented.

DTC was introduced for the first time in 1980s by M.Depenbrock (M. Baishan *et al*, 2009). It is aimed at direct control of torque by controlling the inverter voltage space vector. Stator flux is kept constant and the electromagnetic torque is changed quickly by controlling the stator flux linkage space vector, which in turn is controlled by selection of the appropriate stator voltage generated by the inverter supplying the induction motor (P. Vas, 1998). Hence, the stator flux response is important for ideal performance of induction motor. DTC has less dependence on rotor parameters (B. Zhou *et al*, 2008). It is stator oriented and hence eliminates the need of complicated coordinate transformation (B. Zhou *et al*, 2008), giving the system the benefit of computational reduction. DTC also has advantages of fast torque response, low harmonic losses and low inverter switching frequency (P. Vas, 1998).

The speed regulator generally consists of a PI controller. The PI controller has advantages of simple control system, good adaptability and easiness of design. The basic function of controller is to maintain the output in the desired or required range (P. Lahoty *et al*, 2013). But, the desired performance can be achieved only if the parameters of PI (K_p, K_i) adapt to the system model (B. Zhou *et al*, 2008). The trial and error method of tuning is not very efficient due to the presence of non-linearity in the model of DTC (M. Baishan *et al*, 2009). Therefore, this paper makes use of a bio-inspired technique called PSO which can overcome the inadequacy of the PI controller, thus improving the stator flux response and effectiveness of the DTC system.

2. Control Strategy of DTC

The torque equation for a three phase induction motor in stationary reference frame can be expressed as (M. Baishan *et al*, 2009):

$$T_e = \frac{1}{L_\sigma}\frac{3}{2}\, p\, |\psi_s|\, |\psi_r| \sin\theta \qquad (1)$$

Where ψ_s, ψ_r are stator and rotor fluxes respectively, both referred to the stator reference frame and θ is the angle between the two fluxes. p is the number of pole pairs.

From (1), torque can be controlled either by varying the flux or the angle. But the flux should be maintained constant at the rated value or else there might be problems of saturation or the induction motor might operate at reduced flux. Hence, when the angle θ is varied keeping the stator flux modulus constant, it is called as the direct control of torque.

*Corresponding author: **Trishna Doley**

Figure 1, shows a simplified block diagram of an induction motor drive under DTC (M. Baishan *et al*, 2009).

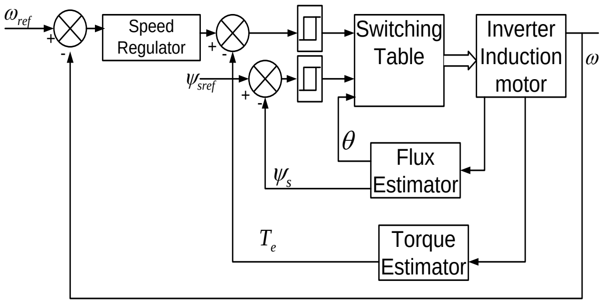

Figure1: Block diagram of Induction Motor

The Induction motor is fed by a three phase voltage source inverter (VSI). The DTC consists of two different hysteresis comparators corresponding to the stator flux and torque. The feedback flux and torque are calculated from machine terminal voltages and currents. The computational block also calculates the stator flux position. The estimated flux and torque are then compared with their reference values and the error produced are fed into their respective hysteresis comparators, which keeps the flux and the torque within their allowed tolerance, based on the following relation (B.K. Bose, 2011):

For flux hysteresis controller,

$H\psi = 1$, for $E\psi > +HB\psi$
$H\psi = -1$, for $E\psi < -HB\psi$

Where, $2HB\psi$ is the total hysteresis- band width of the flux controller.
For torque hysteresis controller,

$H_{Te} = 1$, for $E_{Te} > +HB_{Te}$
$H_{Te} = -1$, for $E_{Te} < -HB_{Te}$
$H_{Te} = 0$, for $-HB_{Te} < E_{Te} < +HB_{Te}$

The stator flux position may be divided in six different sectors each $\pi/3$ angle wide as shown in Figure 2 (M. Baishan *et al*, 2009).The possible dynamic locus of the stator flux along with its different variation depending on the VSI states chosen is also shown.

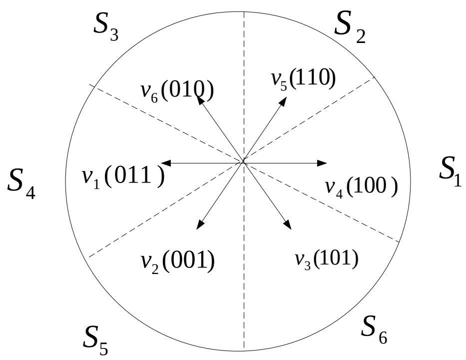

Figure 2: Stator flux vector locus and different voltage vectors

The stator flux estimation equation (M.T. Lazim *et al*, 2011) is:

$$\psi_s = (V_s - R_s i_s)\Delta t \qquad (2)$$

This equation shows that as the stator voltage changes, the stator flux also changes rapidly. But the changes in rotor flux are slow because of its large time constant (B. Purwahyudi *et al*, 2011). Thus the angle between the stator and rotor flux changes and consequently the torque changes.

The output of the two comparators along with the position of the stator flux are fed into the lookup table (B.K. Bose, 2011) (as shown in Table 1), which generates the appropriate control voltage vector for the inverter.

Table 1: Switching table of inverter

H_ψ	H_{Te}	S(1)	S(2)	S(3)	S(4)	S(5)	S(6)
1	1	V_2	V_3	V_4	V_5	V_6	V_1
	0	V_0	V_7	V_0	V_7	V_0	V_7
	-1	V_6	V_1	V_2	V_3	V_4	V_5
-1	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_5	V_6	V_1	V_2	V_3	V_4

The sectors of the stator flux are denoted from S_1 to S_6. The voltage vectors V_0 and V_7 will be selected when the flux and torque errors are within the given hysteresis limit. The DTC based induction motor drive model has been implemented in MATLAB/SIMULINK[R] platform. The Simulink model is as in Figure 3 (M.Baishan *et al*, 2009).

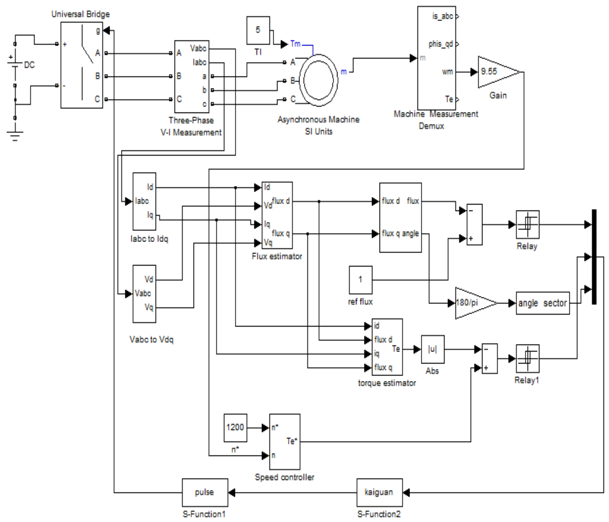

Figure 3: Simulink model of DTC system

3. Particle Swarm Optimization Technique

3.1 PSO BASIC

The PSO is a population-based stochastic search algorithm and is a robust method for solving complex non-linear problems. The algorithm was introduced by James

Kennedy and Russell Eberhart in 1995 (H.F. Rashag *et al*, 2012). The basic idea of this technique was motivated by the simulation of social behavior (B. Zhou *et al*, 2008) of animals such as bird flocking or fish schooling (J. Kennedy *et al*, 1995). It based on the natural process of bird or insect interaction when they flock together in search of food, in a search space without the knowledge of the desirable or best path. But if one of the members finds out the best position, the rest will follow it. Here, each member of the population is called a “Particle” and the population is called a “Swarm”. The search for food starts with a random population moving in arbitrary direction. Each particle while moving through the search space remembers its best previous position along with the best position of its neighbors. The particles share the knowledge of their best position with each other and adjust their own position and velocity derived from the best position of all particles. The particles then keep moving towards better and better position until the swarm reaches an optimum value of fitness function.

The PSO has benefits of algorithmic simplicity, ease of implementation (P. Lahoty *et al*, 2013) and stable convergence to a good solution (H.F. Rashag *et al*, 2012). Hence, it is an ideal choice for optimization problems.

In PSO technique all particles are initiated randomly and evaluated for fitness along with the best position of each particle called “Pbest” and the best value of particle in the population called global best or “Gbest”. To find the optimal solution a loop is started, in which the particles velocity is first updated by Pbest and Gbest value. Then this updated velocity is used to update the current position. When the stopping criteria set in advance is reached, the loop is terminated.

The velocity update equation (B. Zhou *et al*, 2008) for each particle is given by:

$$v_i = \omega v_i + c_1 \times rand_1() \times (p_i - x_i) + c_2 \times rand_2() \times (g - x_i) \quad (3)$$

Where, v_i is the velocity of ith particle at kth iteration. ω is the inertia weight factor. In general, the inertia factor is chosen according to the following equation (B. Zhou *et al*, 2008):

$$\omega = \omega_{max} - \frac{\omega_{max} - \omega_{min}}{K_{max}} \times K \quad (4)$$

Where, K_{max} is the maximum number of iterations and K is the current number of iterations.

c_1 and c_2 is weighting factor or acceleration factor determining the influence of Pbest and Gbest. rand() is any random number between 0 and 1.p_i is the best previous position of ith particle and g is the best particle among all particles in the population.

The current position is updated by the following equation (B. Zhou *et al*, 2008):

$$x_i = x_i + v_i \quad (5)$$

3.2 Implementation of PSO-Pi Controller

Figure 4 (M.M. Eissa *et al*, 2013) shows how PSO-PI controller is implemented in the DTC model.

The input to the PI controller is the error e(t), between the reference speed ω_{ref} and the actual speed ω_r. The PSO algorithm mainly minimizes this error. Hence, the objective function is:

$$e(t) = \omega_{ref} - \omega_r \quad (6)$$

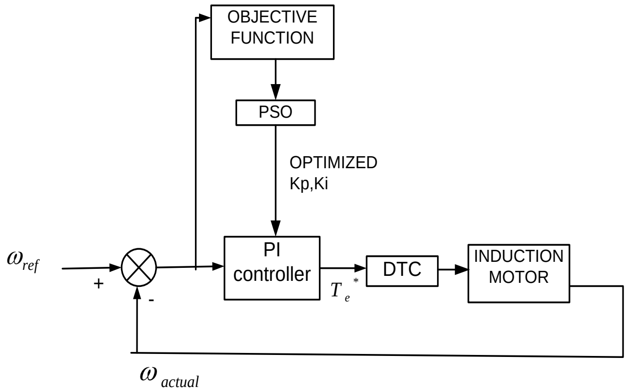

Figure 4: PSO based PI controller of DTC system

Based on this objective function, the PSO optimizes the parameters (k_p, k_i) of the PI controller. The output of the PI controller is the reference torque, which is used to control the DTC system.

The following algorithm describes how PSO is used for optimization purpose.

Step 1: Generation of initial condition.

Set the population size, the acceleration constants c_1 and c_2, inertial factor ω, number of iterations and dimension of the problem.

Step 2: Assessment of searching point of each particle.

Calculate the objective function for each particle. Compare this value with the current Pbest. If this value is better than the current Pbest, then replace the current value with the new value. Also replace the current Gbest with the best value of Pbest, if it is better than the current Gbest. The particle which has the best value is stored.

Step 3: Update each searching point.

The current velocity and position is updated with the new Pbest and Gbest by using (3) and (5) respectively.

Step 4: Checking the iteration stopping criteria.

If the maximum iteration as set in step 1 has reached than exit the loop, otherwise go to step 2 and repeat the process.

Step 5: The particle that has the best value at the end of iteration is the optimal controller parameter. The general flowchart of PSO is shown in Figure 5 (M.M. Eissa *et al*, 2013).

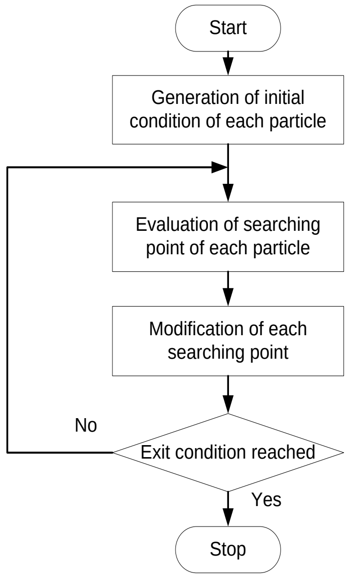

Figure 5: General Flowchart of PSO

4. Simulation Result Analysis

A constant load of 5 Nm is applied with a reference speed of 1200 rpm. The response of stator current and speed with PSO-PI controller is shown in Figure 6 and Figure 7 respectively.

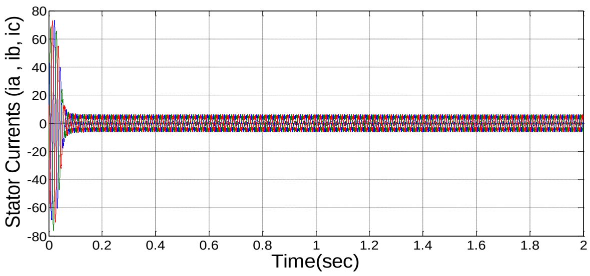

Figure 6: Stator Currents (ia, ib, ic)

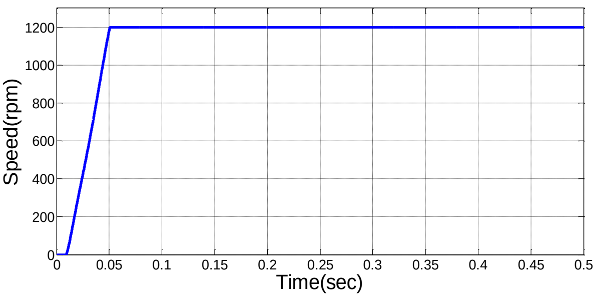

Figure 7: Speed curve with PSO-PI controller

The speed response with a traditional PI controller is shown in Figure 8.

The speed curves plotted as in Figure 7 and Figure 8 shows that the time taken to reach the steady state value is lesser in PSO-PI controller than the traditional PI controller. Also stability is better in the case of PSO-PI controller.

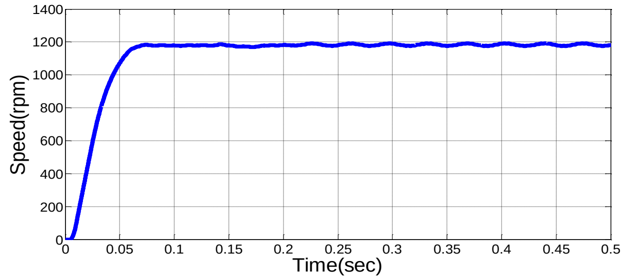

Figure 8: Speed response with traditional PI controller

The stator flux response and plot between d and q axis stator flux with a PSO-PI controller are shown in Figure 9 and Figure 10 respectively.

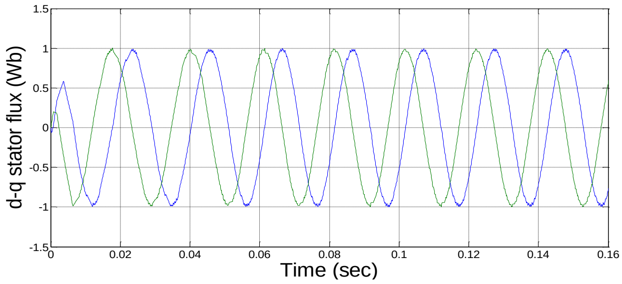

Figure 9: d-q stator flux vs time

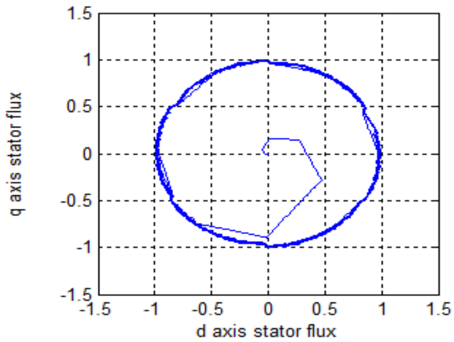

Figure 10: d and q axis stator flux locus

From Figure 9, the d and q axis stator fluxes are sinusoidal in nature and are in the range of (1, -1).

The plot between d and q axis flux as in Figure 10 shows that the stator flux magnitude is maintained constant. This plot is compared with another d-q axis stator flux plot, which is based on Fuzzy Logic controller as in (M. Baishan *et al*, 2009).

A comparative study of performance between the PSO d-q axis stator flux locus as in Figure 10 and Fuzzy Logic controlled d-q axis stator flux locus as in (M. Baishan *et al*, 2009) carried out for the same model and the same parameters shows that in both the cases, the flux linkage increases quickly starting from 0 to the reference value of 1Wb when the motor is running and follows a near circular trajectory due to quadrature relation of d and q axis stator flux. But in the case of PSO optimized PI controlled stator flux locus as in Figure 10, the stability is more as compared to the Fuzzy controlled stator flux locus as in (M. Baishan *et al*, 2009). The comparison thus shows that the PSO-PI controller gives a better Stator Flux response than Fuzzy Logic controller.

Conclusion

The trial and error method of adjusting the PI controller gains in a DTC system is difficult and time consuming. But this difficulty is absent in the PSO based DTC system since PSO requires less computational time and has fast convergence. The PSO based simulation result shows that the stator flux response is quite satisfactory and hence better control over the electromagnetic torque. Thus, the overall performance of the Induction motor drive is improved by using the PSO technique.

Appendix

Induction motor parameters: -

3 KW, 460 Volt, 50 Hz, Rs= 1.115 ohm, Rr =1.083 ohm,
Ls= 0.005974 H, Lm = 0.2037 H, Lr = 0.005974 H, P=2, J= 0.02 kg.m^2.
Control system parameters are:
Flux linkage hysteresis range = 0.005Wb, Reference flux =1 Wb, Torque hysteresis range = 1 Nm.

References

Bose B.K. (2011), Modern Power Electronics and AC Drives, *Pearson Education, Inc., India*.

Purwahyudi B., Suryoatmojo H., Ashari S.M., Hiyama T., (2011), Feed – Forward Neural Network for Direct Torque Control of Induction Motor, *International Journal of Innovative Computing, Information and Control*, Volume 7, Number 11, pp. 6135-6145.

Zhou B., Jing X,F., (2008), Application of Particle Swarm Optimization on DTC for Induction Motors, *IEEE Fourth International Conference on Natural Computation* , Volume 4, pp. 472-476.

Rashag H.F., Koh S.P., Abdalla A.N., Tan N.M.L., Chong K.H., Tiong S.K.., (2012), DTC torque ripple minimization based on PSO-PID controller, *Scientific Research and Essays*, Volume 7, Issue 15, pp. 1564-1572.

Kennedy J., Eberhart R., (1995), Particle Swarm Optimization, *IEEE Int. Conf. on Neural Network* , Volume 4, pp. 1942-1948.

Baishan M., Haihua L., Jinping Z., (2009), Study of Fuzzy control in Direct Torque Control system, *IEEE International Conference on Artificial Intelligence and Computational Intelligence*, Volume 4, pp. 129-132.

Eissa M.M., Virk G.S., AbdelGhany A.M., Ghith E.S., (2013), Optimum Induction Motor Speed Control Technique Using Particle Swarm Optimization, *International Journal of Energy Engineering*, Volume 3, Issue 2, pp. 65-73.

Lazim M.T., Al-khishali M.J.M., Al-Shawi A.I., (2011), Space Vector Modulation Direct Torque Speed Control of Induction Motor, *The 2nd International Conference on Ambient Systems, Networks and Technologies, Procedia Computer Science* , Volume 5, pp. 505-512.

Lahoty P., Girish P., (2013) , A Comparative Study of Tuning of PID Controller using Evolutionary Algorithms, *International Journal of Emerging Technology and Advanced Engineering*, Volume 3, Issue 1, pp. 640-644.

Vas P., (1998), Sensorless Vector and Direct Torque Control, *Oxford University Press, London, UK.*