CHAPTER 4

DESIGN AND SIMULATION OF PI CONTROLLER BASED DRIVE SYSTEM

In this chapter, modeling and simulation of a Proportional Integral (PI)-based control strategy in order to control one of the most important parameters of the IM, viz., the speed, is presented.

4.1 REVIEW ON PI CONTROL STRATEGY

A PI controller is a feedback controller that can control the IM with a weighted sum of the error (difference between the output and desired set-point) and the integral of that value **[24]**. It is a special case of the common PID controller in which the derivative term (D) of the error is not used. The overall block diagram of the speed control of IM using a PI controller is shown in Fig. 4.1.

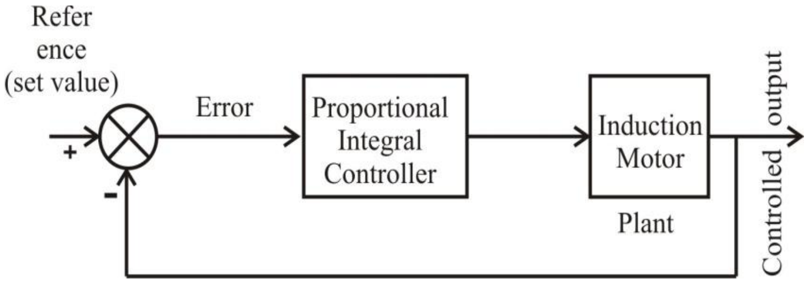

Fig. 4.1: Block diagram of speed control of IM using a PI controller

The PI controller can be expressed in the form of an equation as follows:

$$i_{sq}^{*} = K_p e_{\omega}(t) + K_i \int e_{\omega}(t)dt \qquad (4.1)$$

where i_{sq} is the desired stator q-axis current, $e_{\omega}(t) = \omega_r^{*} - \omega_r$ is the tracking error, K_p is the proportional gain, K_i is the integral gain, $\omega_r{}^{*}$ is the reference speed and ω_r *is* actual speed.

The gains of PI controller are chosen by trial and error. These PI gains are kept constant. But the overshoot for step change of the reference speed and the steady-state error problems arises when the PI controller gains are chosen by trial and error. The closed loop transfer function of the PI speed controller can be written as

$$\frac{\omega_r(s)}{\omega_r^{*}(s)} = \frac{B\left[K_p(s) + K_i\right]}{s^2 + (A + BK_p)s + K_i B} \qquad (4.2)$$

where $\mathrm{A} = \mathrm{D}/\mathrm{J}$, B=K/J.

The characteristic equation of the PI speed controller is given by

$$s^2 + (A + BK_p)s + K_i B = 0 \qquad (4.3)$$

From Equation (4.3), it can be seen that there are two poles of the PI speed controller. If these two poles are chosen equal to be *α,* the gains of the PI controller can be obtained as

$$K_i = \frac{\alpha^2}{B}, \quad K_p = \frac{2\alpha - A}{B} \qquad (4.4)$$

It is seen from the values of K_p and K_i that the proportional gain is proportional to the pole, whereas the integral gain is proportional to the square of the pole **[25]**. Using the gains of the PI controller, the overshoot problem cannot be eliminated. It can be seen that the calculation of the PI controller gains depends on the IM parameters.

The designed PI controller is used in conjunction with the IM plant to control its speed.

4.2 DEVELOPMENT OF THE SIMULINK MODEL

The IM Simulink model with the PI controller was developed using the various toolboxes available in the Matlab/Simulink. The entire system modeled in Simulink is a closed-loop feedback control system consisting of the plants, controllers, feedback systems, mux, de-mux, gain blocks, multipliers, constant blocks, CT and DT blocks, PI editor blocks, clocks, sub-systems, integrators, input sources, work-space blocks, etc.

The developed Simulink model is shown in Fig. 4.2. The specifications of the SCIM used for simulation purposes are given in table 3.1.

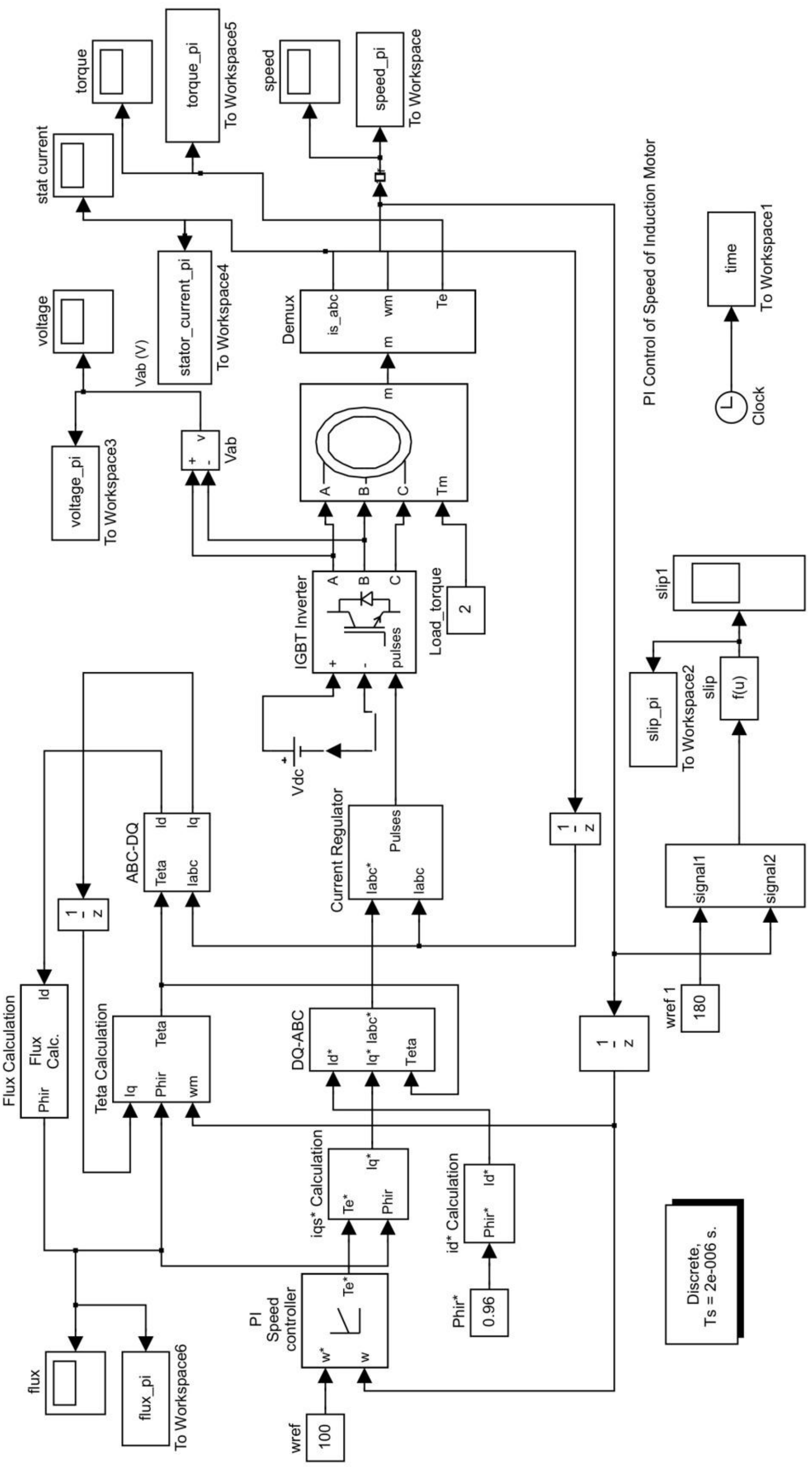

Fig. 4.2: Simulink block diagram of speed control of IM using PI controller

4.3 SIMULATION RESULTS AND DISCUSSIONS

The simulation is run for a period of 3 second in the Simulink environment with a set speed of 100 rads/second, i.e., $\left(\frac{100\times 60}{2\pi}\right)=955$ rpm and with a load torque of 2 *N-m*. The various performance characteristics such as flux, slip, torque, stator current, voltage and speed are observed as shown in Figs. 4.3–4.8, respectively.

From the variation of flux with time as shown in Fig. 4.3, it can be observed that when the motor speed increases (during the transient period), more stator current is required to develop the requisite flux in the air gap, as shown in Fig. 4.6. Hence, the flux also starts increasing during the transient period (0 to 0.5 second) exponentially. Once the motor attains the set rated speed, the flux required to develop the torque almost remains constant after ≥ 0.5 second. Once the flux in the air gap remains constant, the variation of the load torque and speed will not disturb the flux curve. Hence, the IM will be operating at a constant flux.

Fig. 4.4 shows the variation of slip vs. time characteristics for a speed of 100 rad/second (955 rpm). From this simulation result, we infer that the IM attains the set reference speed of 955 rpm in **2 second** using the PI controller.

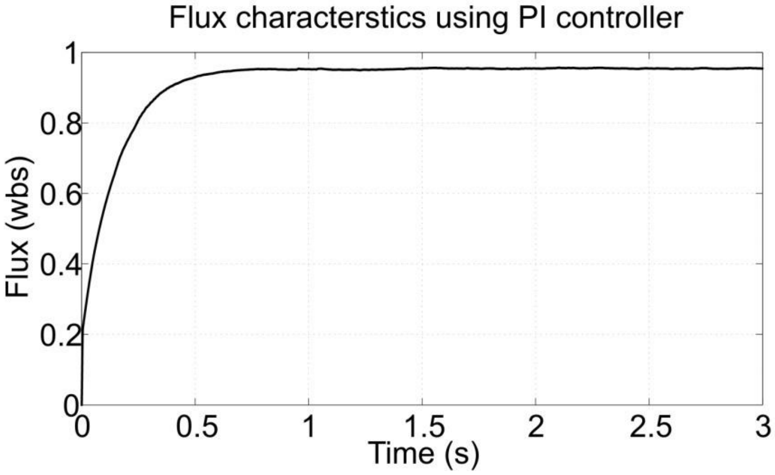

Fig. 4.3: Flux char. using PI

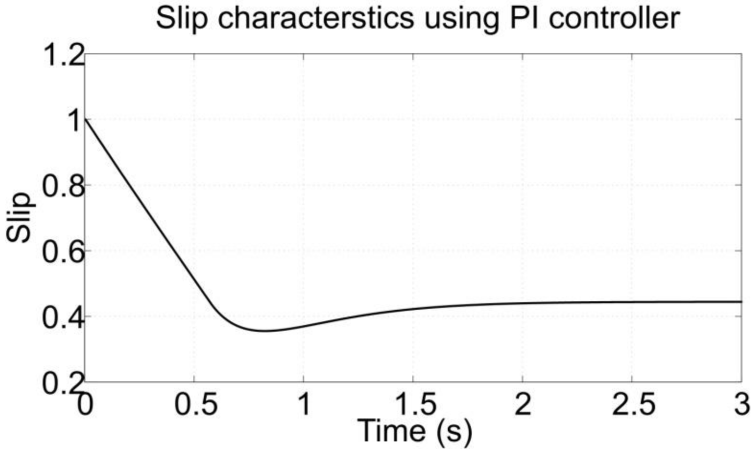

Fig. 4.4: Slip char. using PI

At that instant, the slip of $\frac{N_s - N}{N_s} = \frac{1800 - 955}{1800} = 0.46$ can be verified from the result. Note that the slip decreases from 1.0 to 0.46 linearly in a time span of 2 second. Torque characteristics for a set reference speed of 100 rad/second (955 rpm) are shown in Fig. 4.5. From this, we can deduce that when the motor operates at lower speeds (Fig. 4.8), the slip increases (Fig. 4.4). Hence, the machine requires more torque to attain the set speed. Once the machine reaches the set speed, the average torque of the machine becomes nearly zero after 2 second, which is justified from the torque simulation result in Fig. 4.5.

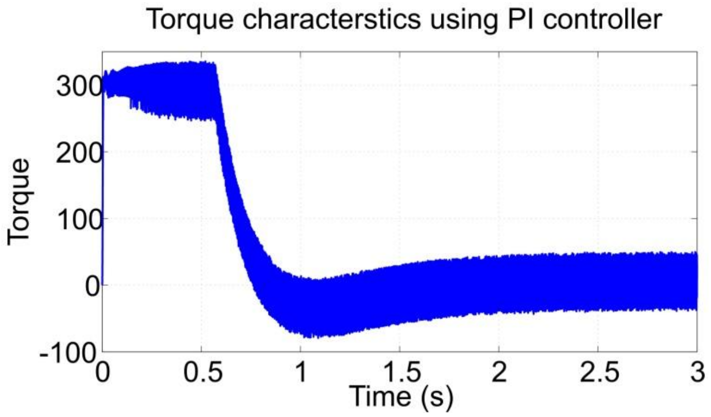

Fig. 4.5: Torque char. using PI

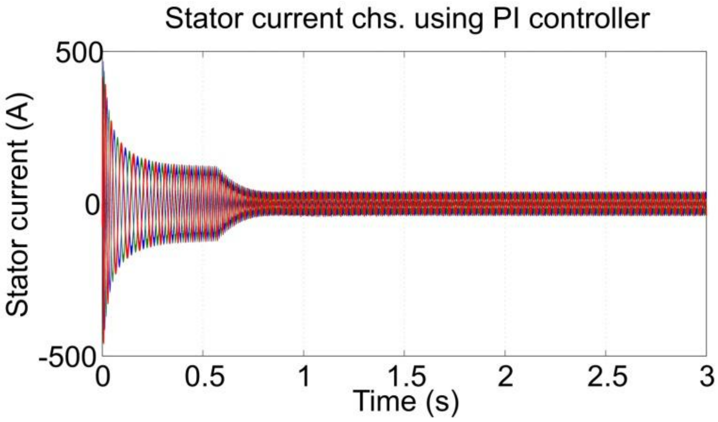

Fig. 4.6: Stator current char. using PI

The variation of the 3Φ stator currents *(i_s)* with t is shown in Fig. 4.6. At lower speeds the slip is more (Fig. 4.4), and the flux required to develop the suitable torque is also more (Fig. 4.3). Moreover, the torque required to reach the set speed is also more (Fig. 4.5). Hence, the magnitude of the stator currents will also be more during the transient periods (starting periods) of the IM. Note that the initial starting transient current is 5–7 times the full load current (steady state current) **[71]**. When the speed reaches the set value from zero, the 3Φ stator currents decrease exponentially. Once it attains the set speed at 2 second, it requires a nominal stator current to drive the IM system.

The terminal voltage of the IM is shown in Fig. 4.7. It is observed from the speed simulation result that by using the PI control, for the set speed of 100 rad/second, the speed reaches its desired set value at **2 second**. The motor speed increases like a linear curve up to the set speed of 955 rpm (100 rad/second) in 0.6 second as shown in Fig. 4.8. After 0.6 second, the response is oscillatory in nature and then finally stabilizes after **2 second**. The reason may be due to the improper selection of the gains in the controller.

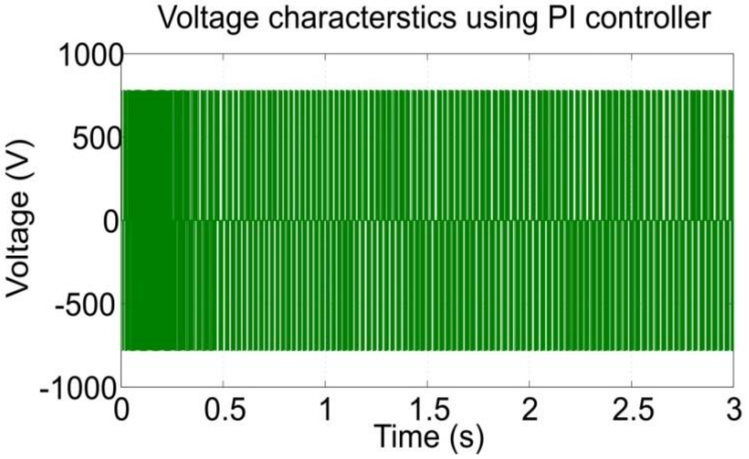

Fig. 4.7: Voltage char. using PI

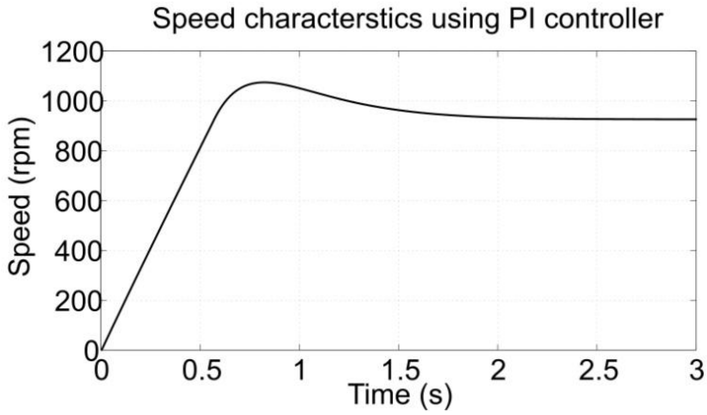

Fig. 4.8: Speed char. using PI

4.4 SUMMARY

A PI based control strategy was developed in this chapter to control the speed of IM. The classical PI controller developed provides control commands for generating the gating signals. The inverter terminal voltage controlled by these gating signals, in turn control the speed of IM drive and has been investigated in this chapter. Simulations were run for a period of **3 second** at set speed of **100 rad/second (955 rpm)** and the results were observed on the corresponding scopes. It is noted that in the PI method, it took **2 second** to reach the set speed. The developed control strategy is simple and reliable as it is the conventional method of control. One of the drawbacks of the designed PI controller is the gains have to be selected properly once the control parameters change. Hence, to overcome the negative aspects of the PI/PID control methods, Mamdani-based FLC methods can be used to design the controller to control the speed of the IM. This is described in the next chapter.